



# **Estudio e implementación de distintos métodos de integración numérica para la resolución de la ecuación de transporte de flujo magnético en la superficie solar**

**Clara Pueyo Ballarín**

**Trabajo de fin de grado en Matemáticas  
Universidad de Zaragoza**

Directores del trabajo:

Luís Rández García

Julia Herrero Albillos

Septiembre 2017



# Prólogo

Desde el inicio de los tiempos el Sol ha sido uno de los astros más estudiados por la Ciencia. Durante cientos de años la mayoría de las civilizaciones fijaron sus ojos en esta estrella, sabiendo que la vida en nuestro planeta dependía de su existencia. Siempre fue intuitiva la necesidad de su luz y calor para la supervivencia, sin embargo hoy sabemos que los efectos del Sol en la Tierra van mucho más allá de lo imaginable.

El campo magnético del Sol abarca todo el Sistema Solar y es fundamental para comprender nuestro medio natural. Lo llamamos Campo Magnético Interplanetario y es, junto con el Campo Magnético Terrestre, corresponsable de muchos de los fenómenos que afectan de forma directa a nuestro planeta. Uno de los más conocidos es la aparición de auroras boreales y australes, visibles desde latitudes cercanas a los polos terrestres. La emisión desde la superficie solar de corrientes de partículas cargadas, denominadas viento solar, ejerce un efecto sobre la superficie exterior de nuestra atmósfera y produce la aparición de manchas lumínicas de distintas tonalidades. Otra de las consecuencias del magnetismo solar es el daño que éste puede causar a nuestros equipos eléctricos o electrónicos. Tanto los satélites con fines científicos como los que tienen un objetivo comercial o militar se ven en ocasiones dañados o incluso destruidos por las tormentas magnéticas provenientes del Sol. Esto puede producir fallos en las señales telefónicas, de televisión, de Internet y en sistemas de posicionamiento global (GPS). Además, también pueden afectar seriamente a las redes eléctricas de las grandes ciudades y producir consecuencias graves en nuestro mundo tecnológico.

En 1859 se produjo la mayor tormenta solar registrada en la historia, denominada evento Carrington. Este fenómeno causó graves daños en el sistema de telégrafo de toda Europa y América del Norte y se llegaron a observar auroras en zonas de latitud media, como Roma (latitud  $41^{\circ}53'30''$  N), Madrid (latitud  $40^{\circ}24'59''$  N) o Montería (Colombia, latitud  $8^{\circ}44'52''$  N). Más recientemente, en 1989 una eyección de plasma solar inutilizó un generador y dejó sin energía eléctrica a 6 millones de personas en la provincia de Quebec (Canadá). Estos acontecimientos son ejemplo de que las consecuencias de una tormenta solar pueden ser muy dañinas. En 2013 el mercado de seguros Lloyd de Londres trató de cuantificar los costes que un fenómeno así podría suponer. Estimó a través de un estudio que los costes de recuperación de una tormenta solar de una potencia elevada, como las dos comentadas anteriormente, podrían alcanzar entre 600.000 y 2.600.000 millones de dólares. Algunos gobiernos también han tomado precauciones ante la posibilidad de sufrir daños por el clima espacial. El 13 de octubre de 2016 Barack Obama, entonces presidente de EEUU, firmó una orden ejecutiva por la cual todas las instituciones estadounidenses (incluidas la NASA, los departamentos de Interior, Energía, Comercio y la Fundación Nacional de Ciencia) debían preparar un plan de acción para antes, durante y después de una tormenta espacial que pudiera afectar a su país. Alegó que un fenómeno de estas características podría causar problemas o incluso anular servicios primarios en continentes completos, como abastecimiento de agua, salud y transporte.

En definitiva, se puede afirmar que las investigaciones que giran en torno al magnetismo solar, especialmente sobre su creación y evolución, son fundamentales y de gran interés en la época en la que vivimos.



# Abstract

This project considers the numerical solution of a partial differential equation that arises in the study of the emergence and transport of magnetic flux through the Sun surface.

To solve numerically this PDE, first of all a semidiscretization in space is done and a set of ordinary differential equations is obtained. After that, different numerical methods to solve the above ordinary differential equation are used.

This research is carried out by Dr. Yeates *et al*, from the Department of Mathematical Sciences of Durham University. I was working with Dr. Yeates as an intern at his department during summer 2016. Personally, I found that period really fascinating, that's why I decided to continue developing my work there as a Final Degree Project.

The ODE has a special structure since it contains a part that depends on a set of Dirac deltas. This fact makes the problem thought-provoking because the solution loses the continuity. Moreover, the estimated error of the numerical values obtained increases immensely at the points where the Dirac delta's values are positive. The Python code for finding a solution of the equation that Dr. Yeates programmed is based on the explicit Euler method. It makes the integration process day by day, but it doesn't show if the results are satisfactory. The main goals are to study if this method is good enough and to improve it by implementing new higher-order methods. Driven by this goals, two Runge-Kutta second-order methods are implemented to estimate the local error of the Euler method. Then, the codes are modified to add a variable stepsize police. Finally, the solution is approximated by programming a third-order three stage explicit Runge-Kutta method, again using variable stepsize, and the results are discussed depending on a tolerance given by the user.

The structure of the project is divided into four different chapters. In the first chapter you can find an introduction to the goals of the project. Some magnetism concepts are explained too in order to understand the topic of the equation. Moreover, it tries to explain how important is researching about the magnetic field of the Sun. Chapter 2 is focused on the equation and its parts. All the variables and some physical phenomenon are defined. Chapter 3 contains the most important mathematical part of the project, that is focused on the analysis and resolution of the equation. It explains why it has a special structure and why is it necessary to approximate the solution numerically. Furthermore, the part that refers to the evaluation of the equation on the code is really important. Then, it studies the different methods and its characteristics, how they are implemented on the initial code and the estimated error of each method, choosing different stepsizes and making them variable, or seeing what happens when changing the fixed tolerance. Finally, in the last chapter all the conclusions are exposed. Also, it explains the difficulties founded and it considers if the code could be better anyway.



# Índice general

<b>Prólogo</b>	<b>III</b>
<b>Abstract</b>	<b>V</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos del trabajo . . . . .	1
1.2. Conceptos previos acerca del magnetismo solar . . . . .	1
<b>2. Modelo de transporte de flujo magnético en la superficie solar</b>	<b>5</b>
2.1. Emergencia (2.5) . . . . .	7
2.2. Transporte . . . . .	7
2.2.1. Difusión (2.2) . . . . .	7
2.2.2. Advección por rotación diferencial (2.1) . . . . .	7
2.2.3. Advección por circulación meridional (2.3) . . . . .	8
2.3. Decaimiento (2.4) . . . . .	8
<b>3. Resolución de la ecuación del modelo</b>	<b>9</b>
3.1. Análisis de la ecuación . . . . .	9
3.2. Evaluación de la ecuación en el código . . . . .	11
3.2.1. Mallado y condiciones iniciales . . . . .	11
3.2.2. Método de diferencias finitas . . . . .	12
3.3. Integración mediante el método de Euler explícito . . . . .	14
3.4. Implementación de métodos Runge-Kutta de orden dos a paso fijo . . . . .	15
3.5. Implementación de métodos Runge-Kutta de orden dos a paso variable . . . . .	18
3.6. Implementación de métodos Runge-Kutta de orden tres a paso variable . . . . .	19
3.7. Resultados proporcionados por el código . . . . .	20
<b>4. Conclusiones</b>	<b>23</b>
4.1. Dificultades encontradas . . . . .	23
4.2. Continuidad y mejoras del trabajo . . . . .	24
<b>Bibliografía</b>	<b>25</b>
<b>ANEXO</b>	<b>27</b>





# Capítulo 1

## Introducción

En este Trabajo de Fin de Grado se pretende estudiar un modelo matemático referente a la actividad magnética solar. Este modelo está siendo desarrollado por un equipo liderado por el Dr. Yeates del Departamento de Ciencias Matemáticas de la Universidad de Durham (Inglaterra), con quien tuve la oportunidad de colaborar durante el verano de 2016 gracias a unas prácticas que ofreció la Universidad de Durham para dicho periodo. Durante ese tiempo con el Dr. Yeates en Inglaterra pude introducirme en el campo del magnetismo solar y en concreto en su investigación, que abarca el estudio de una ecuación en derivadas parciales que modeliza el transporte de flujo magnético en la superficie solar. En particular, comencé a trabajar con un código programado en Python que empleaba como método de integración numérica el método de Euler explícito a paso fijo. Tras las prácticas, tomé la iniciativa de continuar trabajando sobre ese código y tratar de evaluar de forma matemática los resultados obtenidos (ya que no teníamos ninguna información cualitativa de los mismos) y ver si había posibilidades de mejorarlos. De esta experiencia nació el desarrollo de este trabajo, que pretende implementar nuevos métodos de integración numérica para la resolución de la ecuación de transporte de flujo magnético.

### 1.1. Objetivos del trabajo

- Profundizar en el aprendizaje de métodos de integración numérica para la resolución de ecuaciones diferenciales ordinarias, aplicando estos conocimientos a la programación en Python.
- Introducirme en el campo del magnetismo solar y de la investigación científica gracias a mi colaboración en el proyecto del equipo del Dr. Yeates, y realizar de esta forma un trabajo de carácter multidisciplinar.
- Estudiar el proceso de semidiscretización espacial de una ecuación en derivadas parciales y obtener, en el caso de este modelo, un sistema de ecuaciones diferenciales ordinarias con deltas de Dirac. Estudiar la peculiaridad de este tipo de ecuaciones a la hora de ser resueltas mediante métodos de integración numérica.
- Trabajar sobre un código ya existente programado en Python. Obtener datos cualitativos de los resultados obtenidos al resolver la fórmula mediante un método de orden uno a paso fijo (ya implementado por el equipo del Dr. Yeates). Implementar diferentes métodos de orden dos tanto a paso fijo como a paso variable. Implementar un método de orden 3 a paso variable.

### 1.2. Conceptos previos acerca del magnetismo solar

Dado que en este Trabajo de Fin de Grado se va a estudiar un modelo matemático referente a la actividad magnética solar, se introducen brevemente algunos conceptos relacionados con los procesos magnéticos que tienen lugar tanto en el interior como en la superficie de esta estrella, y que por tanto

tomarán parte en el modelo.

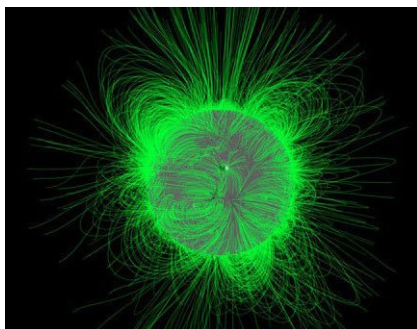


Figura 1.1: Representación de las líneas de flujo. Fuente: NASA

Globalmente, el campo magnético solar tiene una estructura dipolar. El material magnético que se encuentra bajo la superficie del Sol está sometido a continuos movimientos de convección generan la emergencia de flujo magnético (medida del campo magnético total que pasa a través de un área dada) a la misma. Esta emergencia es quasi-periódica y crea a su vez regiones magnéticas dipolares, a las que llamamos regiones activas, que contribuyen a la emisión de destellos solares y eyecciones de masa coronal. Una vez ha salido a la superficie, el flujo magnético se transforma y dispersa por toda ella, y cuando entran en contacto flujos magnéticos de polaridades opuestas se cancelan sus propiedades, anulándose la misma cantidad de flujo de cada polaridad. Estudiar y modelizar todo este proceso de emergencia y transporte de flujo magnético en la superficie solar es importante para conocer y prever el clima espacial y su actividad o repercusión en nuestro planeta.

### MANCHAS SOLARES Y CICLO SOLAR

Las manchas solares son regiones magnéticas que emergen en la fotosfera con temperatura inferior al resto de la superficie de su alrededor, lo que las hace más oscuras y así fácilmente visibles. Estas manchas corresponden a regiones en las cuales el campo magnético es muy intenso. Esto hace que se incremente notablemente la actividad solar a mayor número de manchas solares. Suelen aparecer en parejas, una de cada polaridad, alineadas en la dirección longitudinal del Sol.

Aunque la aparición de manchas en el Sol había sido ya observada desde siglos atrás, son significativos los dibujos realizados en torno a 1612 por Galileo Galilei de las mismas, que fueron un comienzo para estudiarlas de una forma más científica con respecto a las observaciones anteriores. Hoy en día conocemos que el número de manchas solares crece y decrece de forma quasi-periódica cada 11 años, habiendo sido los ciclos más corto y más largo observados de 8 y 14 años respectivamente. Llamamos a esta etapa ciclo solar y en él se aprecia como el número de manchas solares (y por tanto la actividad solar y sus efectos sobre los planetas) va desde un mínimo hasta un máximo, y de nuevo decrece. También cambia la localización de las manchas solares a lo largo de un ciclo solar. En el máximo solar las manchas surgen en lugares más alejados del ecuador solar, repartidas en ambas latitudes: norte y sur. En el mínimo solar, sin embargo, las visualizamos en puntos mucho más próximos al ecuador. El número de manchas solares que podemos apreciar funciona como un marcador visual de la fuerza del campo magnético que emerge del interior solar en un momento preciso dando lugar a eyecciones de masa coronal que afectan a todo el Sistema Solar. Además, cada aproximadamente 11 años se produce una inversión del campo magnético solar. El primer ciclo solar registrado fue en 1755, por lo que actualmente nos encontramos en el vigésimo cuarto ciclo solar.

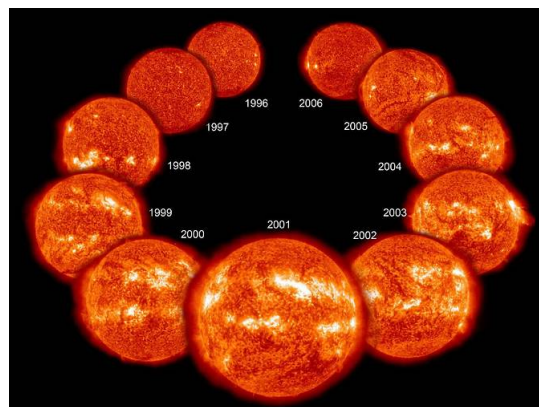


Figura 1.2: Ciclo Solar 23. Fuente: NASA

## DIAGRAMAS DE MARIPOSA

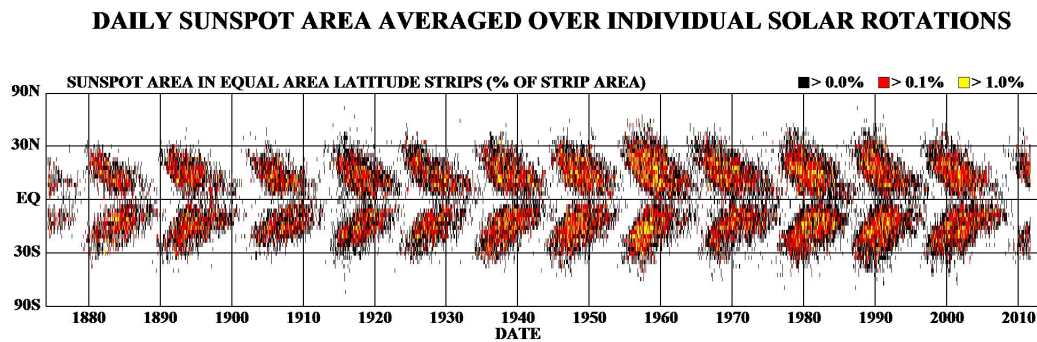


Figura 1.3: Diagrama de mariposa. Fuente: NASA

En la figura 1.3 se muestra un diagrama de mariposa, que representa la localización (latitud) de las manchas solares en cada instante de tiempo. El nombre de este diagrama viene dado por su apariencia. La línea central, de izquierda a derecha, es el ecuador del Sol, y las manchas ocurren al norte y sur, sobre y debajo de esta línea. Los colores representan el área que abarcan las manchas. En este diagrama se demuestra de una forma visual que a iguales latitudes hacia el norte y hacia el sur, aparecen aproximadamente el mismo número de manchas solares. El máximo solar corresponde a los puntos donde las manchas están más alejadas del ecuador, y el mínimo a los puntos en los que están más próximas al mismo. Además, también se aprecia que los ciclos solares son quasi-periódicos, pues se va repitiendo la imagen de una forma similar, aunque no en todos los ciclos el campo magnético llega a tener la misma intensidad. En definitiva, son una herramienta visual para estudiar la evolución de un ciclo solar.



## Capítulo 2

# Modelo de transporte de flujo magnético en la superficie solar

En 1961 se produjo un avance significativo en el entendimiento de la naturaleza de las manchas solares. Babcock presentó su teoría acerca de la topología del campo magnético en la superficie solar [2]. En ella explicaba la deformación del propio campo magnético debida a la rotación diferencial (el Sol rota sobre sí mismo, pero al estar compuesto de gas y plasma, no lo hace en todos sus puntos a la misma velocidad angular) y describía cómo la emergencia de flujo magnético a la superficie forma grupos dipolares de manchas solares. Babcock estimó que la aparición de estas regiones activas se producía de forma periódica, dando lugar a ciclos solares de 22 años. Su teoría lograba explicar la información reflejada por los diagramas de mariposa y la polaridad que se había observado en el Sol.

En 1964, partiendo de los resultados de Babcock, Leighton presentó su teoría sobre el transporte de flujo magnético en la superficie solar [1]. Según Leighton, una vez las regiones magnéticas han emergido a la superficie se expanden y se difunden por la misma. Lo hacen siguiendo un mecanismo físico basado en un intercambio de energía entre las componentes poloidal y toroidal del campo magnético. Tras esta dispersión de regiones magnéticas dipolares, llega su decaimiento, mostrando un patrón de inclinación respecto de la dirección este-oeste. Todo este proceso termina por neutralizar y a continuación invertir la polaridad en el Sol cada aproximadamente 11 años, lo que propone la reducción de ciclo solar a 11 años, y lo divide en cinco fases diferentes.

Los modelos que parten de los estudios de Babcock y Leighton son exitosos en el campo teórico de la física solar y aún hoy continúan desarrollándose. El modelo estudiado en este trabajo parte de sus bases para estudiar el transporte y la dispersión de flujo magnético en la superficie solar y que será detallado a continuación.

La mayoría de los estudios previos han utilizado métodos de extrapolación del potencial para conseguir una primera aproximación de los movimientos de transporte de flujo y de la *Heliospheric Current Sheet*<sup>1</sup>, pero sus predicciones de los destellos solares y las eyecciones de masa coronal son muy limitadas. Esto causa que dichos modelos no consideren la energía libre, y la escasez de corrientes eléctricas simuladas no permite estimar el flujo abierto a todo el campo magnético solar, en especial en los periodos más activos. Como consecuencia, no se pueden obtener resultados suficientemente precisos acerca del clima solar en un instante de tiempo. Además, diferentes códigos numéricos de extrapolación han dado lugar a resultados distintos, lo que supone un problema grave. Por otra parte, en los últimos años

---

<sup>1</sup>La Heliospheric Current Sheet es la superficie ilusoria que determina el cambio de polaridad del campo magnético a lo largo de todo el Sistema Solar. Parte del ecuador y la inclinación de los ejes magnéticos, relacionados con la inclinación del eje de rotación del Sol, hace que vaya adquiriendo la forma de una bandera moviéndose con el viento conforme se aleja del Sol.

se han llevado a cabo modelos magnetohidrodinámicos <sup>2</sup> que han resultado ser prácticos. Sin embargo, todavía no son capaces de simular la evolución temporal de la corona más allá de varios meses.

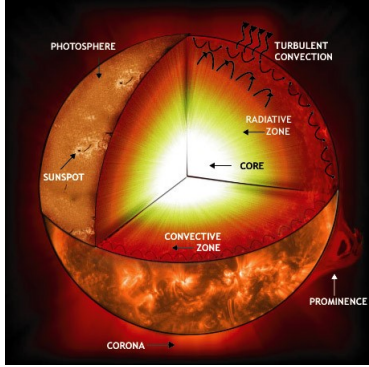


Figura 2.1: Estructura interna del Sol. Fuente: Fundamentos de astrofísica, Jorge I. Zuluaga, Universidad de Antioquia.

Los avances del modelo estudiado en este TFG, que actualmente continua desarrollándose por el Dr. Yeates, parten de considerar la estructura magnética solar y su topología mediante simulaciones del campo magnético en la corona en función del tiempo, es decir, en cada momento de su evolución histórica. Para ello se utiliza una aproximación simplificada de la evolución real de la corona y así quedan atrás las extrapolaciones de datos en la fotosfera. También se simula la emergencia de flujo y sus movimientos de forma continua. Así, este modelo permite predecir el clima espacial porque considera los movimientos de la energía libre y las corrientes eléctricas, lo que parece resolver los problemas de los modelos anteriores.

Se presenta a continuación la ecuación diferencial que describe la evolución temporal del campo magnético radial  $B(\theta, \phi, t)$  en la superficie solar,

$$\frac{\partial B}{\partial t} = -\omega(\theta) \frac{\partial B}{\partial \phi} \quad (2.1)$$

$$+ \frac{\eta}{R^2} \left[ \frac{1}{\sin \theta} \frac{\partial}{\partial \theta} \left( \sin \theta \frac{\partial B}{\partial \theta} \right) + \frac{1}{\sin^2 \theta} \frac{\partial^2 B}{\partial \phi^2} \right] \quad (2.2)$$

$$- \frac{1}{R \sin \theta} \frac{\partial}{\partial \theta} (v(\theta) \sin \theta B) \quad (2.3)$$

$$- \frac{1}{\tau} B \quad (2.4)$$

$$+ S(\theta, \phi, t), \quad (2.5)$$

donde  $\theta$  y  $\phi$  son las coordenadas esféricas (colatitud y longitud respectivamente),  $t$  el tiempo,  $\omega(\theta)$  la velocidad angular,  $\eta$  la constante de difusión,  $R$  el radio solar,  $v(\lambda)$  la velocidad,  $\tau$  la constante de decaimiento y  $S(\theta, \phi, t)$  el sumando correspondiente a la emergencia de nuevas regiones magnéticas [5].

Esta ecuación incorpora cinco efectos físicos que serán explicados a lo largo del capítulo y que responden a cada uno de los cinco sumandos que la componen:

- Emergencia (2.5)

- Transporte

- Difusión (2.2)

- Adección por rotación diferencial (2.1)

- Advección por circulación meridional (2.3)

- Decaimiento (2.4)

<sup>2</sup>La magnetohidrodinámica (MHD) estudia la dinámica de los fluidos que son buenos conductores de la electricidad, y específicamente los efectos que aparecen por la interacción entre el movimiento del fluido y un campo magnético cualquiera que pueda estar presente. (Universidad de Navarra, <http://fisica.unav.es/mhd/>).

## 2.1. Emergencia (2.5)

Este modelo asume que la emergencia de regiones magnéticas dipolares ocurre en un periodo de tiempo muy corto, lo que permite tomarla como un fenómeno instantáneo. En la ecuación esto se refleja en el sumando (2.5), es decir, en el término  $S(\theta, \phi, t)$  que viene dado por:

$$S(\theta, \phi, t) \equiv S(\theta, \phi) \delta(t - t_i) \quad (2.6)$$

Esto significa que el valor de  $S$  será no nulo únicamente en los instantes de tiempo  $t_i$  en los que produzca la emergencia de flujo magnético a la superficie solar. Además, se asume que esta emergencia se produce de forma aislada (será para cada  $t_i$  y para cada par de coordenadas  $(\theta, \phi)$  una constante determinada) y cada región dipolar corresponde a la emergencia de un par de flujos de distinta polaridad pero de igual intensidad entre sí.

## 2.2. Transporte

El transporte de flujo magnético, como se verá a continuación, puede considerarse de dos formas diferentes: como movimientos aleatorios a pequeña escala (difusión) o como flujos con un comportamiento sistemático (advección).

### 2.2.1. Difusión (2.2)

El sumando de la ecuación (2.2) modeliza la difusión de la componente radial del campo magnético. Para ello, se toma una constante de difusión  $\eta$  que permite basar esta modelización globalmente.

La estimación inicial del valor de esta constante de difusión fue dada por Leighton (1964), que basándose en el tiempo que transcurría hasta la inversión de polaridad, determinó que debería tomar un valor entre 770 y 1540 km<sup>2</sup>s<sup>-1</sup>. Sin embargo, este valor se consideró menor en 1977, cuando Mosher estudió el periodo de vida de las regiones magnéticas activas a través de datos proporcionados por diferentes observaciones. Se consideró entonces que el valor de  $\eta$  estaría entre los 200 y los 600 km<sup>2</sup>s<sup>-1</sup>. A partir de entonces, se han llevado a cabo múltiples estudios para determinar el mejor valor de la constante de difusión, aunque cada modelo considera un valor diferente.

Para este trabajo se considera el valor de la constante de difusión  $\eta = 500 \text{ km}^2\text{s}^{-1}$ , tal y como detalla el modelo trabajado por el Dr. Yeates.

### 2.2.2. Advección por rotación diferencial (2.1)

Esta clase de movimientos se producen en dirección longitudinal. El modelo representa en el sumando (2.1) el transporte de flujo en cada pequeña zona de la superficie solar, teniendo en cuenta que, como la composición del sol es plasmática y gaseosa, no rota uniformemente en toda la superficie.

Los movimientos de flujo magnético en la superficie solar por rotación diferencial fueron los primeros en ser conocidos (Christof Scheiner, 1610), lo que hace que en la actualidad hayan podido ser desarrollados con más detalle. Destacan las investigaciones de Newton y Nunn (1951) y de Ward (1966) que determinaron que cuanto menor era el tamaño de una mancha solar, mayor era su velocidad de rotación, aunque sólo describieron estos movimientos en latitudes inferiores a 30°. Posteriormente los estudios de Howard y Harvey (1970), Howard et al. (1984), Snodgrass et al. (1984) y Ulrich et al. (1988) extendieron esta descripción a todas las latitudes. Recientemente, se ha ido ampliando el conocimiento de estos movimientos, destacando los avances de Thompson et al. (1996), Schou et al. (1998), Corbard y Thompson (2002), Komm et al. (2003), Meunier (2005), Hathaway y Rightmire (2010-2011) y Howe (2011). Hoy en día se conoce que la superficie de rotación diferencial varía sistemáticamente durante

cada ciclo solar. [5]

En el modelo considerado en este trabajo se tomará la velocidad angular  $\omega(\theta)$  como:

$$\omega(\theta) = 0,521 - 2,396 \cos^2(\theta) - 1,787 \cos^4(\theta) \text{ deg dia}^{-1}$$

### 2.2.3. Advección por circulación meridional (2.3)

Este movimiento se produce en dirección latitudinal. En el sumando (2.3) de la ecuación se modela el transporte del flujo magnético de la superficie solar hacia los polos, donde se produce la inversión de polaridad de los campos magnéticos. El transporte del mismo de nuevo hacia el ecuador se produce en el interior del Sol, y una vez se encuentra en la zona ecuatorial se cancela con el flujo de distinta polaridad procedente del otro hemisferio.

Los movimientos de advección por circulación meridional dependen de la velocidad  $v(\lambda)$ , que en este modelo viene dada según las conclusiones de Schüssler y Baumann en 2006 [3]:

$$v(\lambda) = -v_0 \sin(2\lambda) \exp(\pi - 2|\lambda|) \text{ ms}^{-1}$$

siendo  $\lambda = \pi/2 - \theta$  la latitud y  $v_0 = 16 \text{ ms}^{-1}$  una constante<sup>3</sup>.

## 2.3. Decaimiento (2.4)

Si no fuera por la emergencia de nuevo flujo magnético modelada por (2.5), el flujo magnético total en la superficie solar se iría reduciendo de forma monótona. Al aproximarse regiones magnéticas dipolares entre sí, se anula cierta cantidad de flujo entre polaridades opuestas. Este proceso es más rápido cuanto más próximas están unas de otras, pero incluso entre los flujos que se encuentran en los polos solares se produce esta anulación.

En nuestro modelo, este fenómeno viene representado por el sumando (2.4), donde se fija una constante de decaimiento  $\tau = 10$  años.

---

<sup>3</sup>Aunque suele tomarse  $v_0$  como el máximo valor de  $v(\lambda)$ , en el caso del modelo estudiado por el Dr. Yeates se fija esta constante para definir dicho término.



## Capítulo 3

# Resolución de la ecuación del modelo

Uno de los métodos más utilizados para la resolución numérica de una ecuación en derivadas parciales consiste en mantener la variable en tiempo continua y semidiscretizar las variables espaciales (por ejemplo con el método de las diferencias finitas). En este caso se trata de una semidiscretización espacial y el problema semidiscretizado se reduce a un sistema de ecuaciones diferenciales ordinarias cuya dimensión coincide con el número de nodos espaciales que tenga la discretización espacial.

### 3.1. Análisis de la ecuación

La emergencia de flujo magnético a la superficie solar se ha modelizado como una secuencia de regiones magnéticas dipolares discretas. Gracias a estudios estadísticos y a observaciones llevadas a cabo durante los pasados años se cuenta con un registro de datos<sup>1</sup> que simula dichas emergencias. En él, se proporciona la información del día de aparición de cada una de estas regiones singulares, sus coordenadas (longitud y latitud), la separación entre los polos de cada polaridad (lo que nos informa a su vez del tamaño de la región magnética), el valor del flujo magnético de cada polaridad y el ángulo de inclinación con respecto a la línea este-oeste. Todos estos datos permiten obtener una constante  $\alpha$  que represente la emergencia de flujo en cada instante y en cada localización en la que ésta tenga lugar, aunque para este trabajo se omitirá este cálculo.

Para modelizar cada emergencia en el instante de tiempo  $t_i$  se utiliza la fórmula (2.6), por lo que hay que añadir al sistema de ecuaciones diferenciales un término del tipo  $\sum S(\theta, \phi) \delta(t - t_i)$ . La solución de este tipo de ecuaciones diferenciales se explica en [8].

**Proposición 3.1.** Sean  $f$  y  $g$  globalmente Lipschitz (uniformemente en intervalos de tiempo compactos), y sea  $G(y) = \int dy/g(y)$  invertible. Entonces las soluciones de la ecuación

$$\begin{cases} y'(t) = f(t, y(t)) + g(y(t)) \delta(t - t_1) \\ y(t_0) = y_0 \end{cases} \quad (3.1)$$

Pueden escribirse como

$$\bar{y}'(t) = \begin{cases} \bar{y}_1'(t), & t \leq t_1 \\ \bar{y}_2'(t), & t > t_1 \end{cases}$$

Donde  $\bar{y}_1$  es solución de

$$\begin{cases} \bar{y}_1'(t) = f(t, \bar{y}_1(t)) \\ \bar{y}_1(t_0) = y_0 \end{cases}$$

---

<sup>1</sup>Esta base de datos ha sido proporcionada por el Dr. Yeates del Departamento de Ciencias Matemáticas, Universidad de Durham (Inglaterra).

e  $\bar{y}_2$  es solución de

$$\begin{cases} \bar{y}_2'(t) = f(t, \bar{y}_2(t)) \\ \bar{y}_2(t_1) = G^{-1}(G(\bar{y}_1(t_1)) + 1) \end{cases}$$

En el caso del modelo estudiado, con  $t \in [t_0, T]$ , el sumando con una delta de Dirac en la ecuación (3.1) corresponde a  $S(\theta, \phi, t)$  en la ecuación (2.1)-(2.5). Además, como se ha explicado al inicio del capítulo,  $g(y(t))$  es la constante  $\alpha$  dada por los días de emergencia, y por tanto  $G(y)$  queda definida por:

$$G(y(t)) = \int \frac{dy}{\alpha} = \frac{1}{\alpha}y(t)$$

Así, la función  $\bar{y}_2$  viene dada por:

$$\begin{cases} \bar{y}_2'(t) = f(t, \bar{y}_2(t)) \\ \bar{y}_2(t_1) = \alpha(\frac{1}{\alpha}\bar{y}_1(t_1) + 1) = \bar{y}_1(t_1) + \alpha \end{cases}$$

Y finalmente, aplicando la proposición anterior, la solución de la ecuación del modelo converge a la solución de la ecuación definida a trozos:

$$\begin{cases} \bar{y}_1'(t) = f(t, \bar{y}_1(t)) & t \in [t_0, t_1] \\ \bar{y}_1(t_0) = y_0 \end{cases} \leftrightarrow \begin{cases} \bar{y}_2'(t) = f(t, \bar{y}_2(t)) & t \in [t_1, T] \\ \bar{y}_2(t_1) = \bar{y}_1(t_1) + \alpha \end{cases}$$

Notar que por tanto, en un problema de estas características, el efecto producido es una discontinuidad de salto finito en  $t = t_1$ . En la figura 3.1 se puede observar esta discontinuidad para un ejemplo concreto.

### Ejemplo 3.1.

$$\begin{cases} y'(t) + y(t) = (t+1)\delta(t-1) & t \in [t_0, 2] \\ y(0) = 2 \end{cases}$$

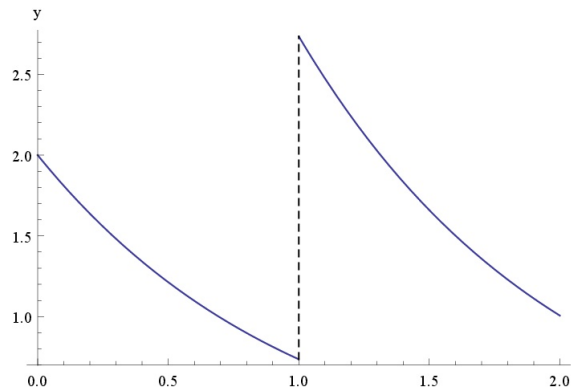


Figura 3.1: Gráfica de la solución del ejemplo 3.1.

Por otra parte, en el caso del modelo estudiado hay varios instantes  $t_i$  en los que la función tomará un valor diferente. Además, el valor de la constante  $\alpha$  será distinto para cada punto  $t_i$ :

$$y' = f(y) + \sum_{i=1}^N \delta(t - t_i) \alpha(t) \quad (3.2)$$

La parte del sumatorio, de nuevo con una delta de Dirac, hace referencia a la aparición de emergencias en los instantes  $t_i$ . Así, en estos puntos en los que, según los datos facilitados, surge una nueva región magnética, se perderá la continuidad de la solución.

La complejidad de la ecuación, la alta dimensión y el número elevado de deltas de Dirac de la función derivada correspondiente al modelo estudiado impiden que pueda resolverse la ecuación aplicando la transformada de Laplace. Es por ello que deben utilizarse métodos numéricos de integración.

### 3.2. Evaluación de la ecuación en el código

Para la integración de la ecuación (2.1)-(2.5) es conveniente introducir un vector potencial  $(A_\theta, A_\phi)$ , que viene dado por las siguientes ecuaciones:

$$\frac{\partial A_\theta}{\partial t} = \omega(\theta)R \sin(\theta)B - \frac{\eta}{R \sin(\theta)} \frac{\partial B}{\partial \phi} - \frac{1}{\tau} A_\theta + S_A(\theta, \phi, t) \quad (3.3)$$

$$\frac{\partial A_\phi}{\partial t} = -v(\theta)B + \frac{\eta}{R} \frac{\partial B}{\partial \theta} - \frac{1}{\tau} A_\phi \quad (3.4)$$

El término  $S_A$  es el que corresponde a las regiones magnéticas dipolares emergentes y su valor en cada día de emergencia y en cada par de coordenadas se tomará constante. En este trabajo se omitirá el cálculo de estos valores de  $S_A$ , que vendrán dados por los datos proporcionados, tal y como se explica en la subsección 3.1. Además, este termino guarda la siguiente relación con el término (2.5):

$$S(\theta, \phi, t) \equiv -\frac{\partial S_A}{\partial \phi}(\theta, \phi, t)$$

También ha sido proporcionado un código inicial<sup>2</sup>, programado en Python, que aproxima la solución de la ecuación (2.1)-(2.5). En él se trabaja sobre este vector, que permite describir el campo magnético radial  $B(\theta, \phi, t)$  de la siguiente forma:

$$B(\theta, \phi, t) = \frac{1}{R \sin(\theta)} \left( \frac{\partial}{\partial \theta} (\sin \theta A_\phi) - \frac{\partial A_\theta}{\partial \phi} \right) \quad (3.5)$$

Será objetivo de este TFG analizar ese código inicial e implementar distintos métodos de integración numérica para tratar de mejorar la resolución de la ecuación (2.1)-(2.5).

#### 3.2.1. Mallado y condiciones iniciales

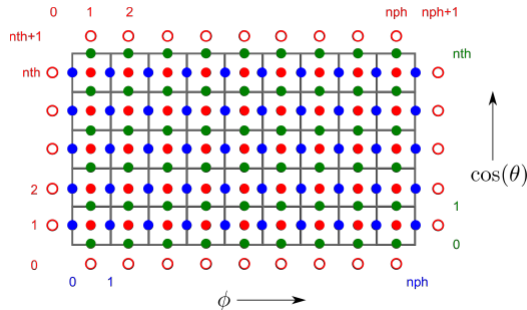


Figura 3.2: Mallado del dominio.

Siendo  $\phi$  y  $\theta$  la longitud y colatitud respectivamente, se construye una partición de toda la superficie solar en celdas de las mismas dimensiones según  $\phi$  y  $\cos(\theta)$ , ya que esta elección del mallado simplificará posteriormente la evaluación de la función. Se toman  $nph = 180$  celdas en longitud ( $\phi \in (0, 2\pi)$ ) y  $nth = 90$  celdas en latitud ( $\cos \theta \in (-1, 1)$ ).

En la figura (3.2) se representan estas celdas de forma esquemática. Los puntos azules hacen referencia a los centros en los ejes verticales, donde se define  $A_\theta$ , y análogamente los verdes a los centros en los ejes horizontales, donde se define  $A_\phi$ . Así, aplicando

el método de las diferencias finitas en la evaluación de estas componentes se permite estimar el valor de

<sup>2</sup>Dr. Yeates, Departamento de Ciencias Matemáticas, Universidad de Durham (Inglaterra).

$B$ , que está definido en los puntos rojos, es decir, en los centros de cada celda.

Los puntos no coloreados pertenecen a celdas imaginarias, en las que se define el valor de  $B$  de la siguiente forma:

Para cada valor de la longitud (ver ecuación (3.3)),  $B$  es un valor modular, es decir, para cada valor de  $\cos(\theta)$  el valor de  $B$  en la celda 0 será el valor de  $B$  en la celda  $nph$  y el valor de  $B$  en la celda  $nph + 1$  será el valor de  $B$  en la celda 1.

Para cada valor de la colatitud (ver ecuación (3.4)), para asegurar que no haya difusión más allá de los límites de la celda, se fija en los ejes el valor de la velocidad como  $v = 0$  y  $\frac{\partial B}{\partial \theta} = 0$ . A partir de ahí, para cada valor de  $\phi$  el valor de  $B$  en la celda 0 será el valor de  $B$  en la celda 1, y el valor de  $B$  en la celda  $nth + 1$  será el valor de  $B$  en la celda  $nth$ .

Sea  $\lambda = \frac{\pi}{2} - \theta$  la latitud y  $B_0$  una constante que representa la intensidad inicial del campo magnético polar, tomando el flujo meridional particular  $v_0$  dado por  $v(\lambda)$  en la subsección 2.2.3, se define la condición inicial para  $B(\lambda, \phi, t_0 = 0)$  como:

$$B(\lambda, \phi, 0) = B_0 \frac{\lambda}{|\lambda|} \exp \left[ - \frac{v_0 R \exp(\pi - 2|\lambda|)}{4\eta} (\sin(2|\lambda|) + \cos(2|\lambda|)) \right]. \quad (3.6)$$

Esta expresión es una solución estacionaria de la ecuación que resulta al considerar en la ecuación del modelo únicamente los sumandos referentes al transporte de flujo magnético, es decir, (2.1), (2.2) y (2.3), despreciando los referentes a la emergencia (2.5) y decaimiento (2.4).

Por otra parte, dada la ecuación (3.5), se toman las siguientes condiciones iniciales para  $A_\theta$  y  $A_\phi$ :

$$A_\phi = 0 \quad (3.7)$$

$$A_\theta = -\sin \theta \int_0^\phi B(\theta, \hat{\phi}, 0) d\hat{\phi} \quad (3.8)$$

### 3.2.2. Método de diferencias finitas

Se trata de estimar el valor de  $B$  en la celda  $(i, j)$ , siendo  $i$  y  $j$  los números de la columna y la fila de la celda en la malla respectivamente (ver numeración en rojo en la figura 3.2). Así, denotaremos por  $\theta_j$  y  $\phi_i$  a los valores de dichos ángulos en cada celda.

En primer lugar se estimarán los valores de  $\frac{\partial A_\theta}{\partial t}$  y  $\frac{\partial A_\phi}{\partial t}$  en los ejes correspondientes de la malla. Para ello, se deben tomar los valores de  $B$ ,  $\frac{\partial B}{\partial \phi}$  y  $\frac{\partial B}{\partial \theta}$  adecuados para evaluar las ecuaciones (3.3) y (3.4). Se denotará como  $i_e$  a los ejes verticales y  $j_e$  a los ejes horizontales (ver numeración en azul y en verde respectivamente en la figura 3.2).

Se explica a continuación el proceso de evaluación de la ecuación (3.3).  $A_\theta$  está definida en los ejes verticales  $(i_e, j)$  de la malla, por lo que se toma el valor de  $B$  en dos celdas longitudinalmente consecutivas y se calcula su valor en su punto central. En este caso se trabaja con matrices de dimensiones  $(nph + 1) \times nth$ , es decir,  $181 \times 90$ .

Para  $i \in [0, nph + 1]$ ,  $i_e \in [0, nph]$ ,  $j \in [1, nth]$ :

$$B_{(i_e, j)} = \frac{B_{(i, j)} + B_{(i+1, j)}}{2}$$

$$\frac{\partial B}{\partial \phi}(i_e, j) = \frac{B_{(i+1,j)} - B_{(i,j)}}{\phi_{i+1} - \phi_i} = \frac{B_{(i+1,j)} - B_{(i,j)}}{\phi_1 - \phi_0},$$

y sustituyendo en la ecuación (3.3):

$$\frac{\partial A_\theta}{\partial t}(i_e, j) = \omega(\theta)R \sin(\theta) \left[ \frac{B_{(i,j)} + B_{(i+1,j)}}{2} \right] - \frac{\eta}{\sin(\theta)} \left[ \frac{B_{(i+1,j)} - B_{(i,j)}}{\phi_1 - \phi_0} \right] - \frac{1}{\tau} A_\theta + S_A(\theta, \phi, t) \quad (3.9)$$

Si  $t_i$  es un día donde se produce una emergencia, el valor de  $S_A(\theta, \phi, t) \equiv S_A(\theta, \phi) \delta(t - t_i)$ .

La ecuación (3.4) se evalúa de forma análoga.  $A_\phi$  está definida en los ejes horizontales  $(i, j_e)$  de la malla, y en este caso trabajaremos con matrices de dimensiones  $nph \times (nth + 1)$ , es decir,  $180 \times 91$ . Empleando la relación:

$$\frac{\partial B}{\partial \theta} = -\sin \theta \frac{\partial B}{\partial \cos \theta},$$

para  $i \in [1, nph]$ ,  $j \in [0, nth + 1]$ ,  $j_e \in [0, nth]$ :

$$B_{(i,j_e)} = \frac{B_{(i,j)} + B_{(i,j+1)}}{2}$$

$$\frac{\partial B}{\partial \theta}(i, j_e) = -\sin \theta \frac{B_{(i,j+1)} - B_{(i,j)}}{\cos \theta_j - \cos \theta_{(j-1)}} = -\sin \theta \frac{B_{(i,j+1)} - B_{(i,j)}}{\cos \theta_1 - \cos \theta_0}.$$

Finalmente, sustituyendo en la ecuación (3.4):

$$\frac{\partial A_\phi}{\partial t}(i, j) = -v(\theta) \left[ \frac{B_{(i,j)} + B_{(i,j+1)}}{2} \right] - \frac{\eta}{R} \sin \theta \left[ \frac{B_{(i,j+1)} - B_{(i,j)}}{\cos \theta_1 - \cos \theta_0} \right] - \frac{1}{\tau} A_\phi \quad (3.10)$$

Una vez se hayan obtenido las estimaciones numéricas para el vector potencial, se volverá a aplicar el método de las diferencias finitas para evaluar  $B$  en el centro de las celdas. Se utilizará para ello una expresión equivalente a (3.5):

$$B(\theta, \phi, t) = -\frac{1}{R} \frac{\partial}{\partial \cos(\theta)} (\sin(\theta) A_\phi) - \frac{1}{R \sin(\theta)} \frac{\partial A_\theta}{\partial \phi} \quad (3.11)$$

La matriz de valores de  $B$  tiene dimensiones  $(nph + 2) \times (nth + 2)$ , es decir,  $182 \times 92$ . Para evaluar esta expresión se vuelve a aplicar el método de las diferencias finitas, de forma análoga a los procesos de evaluación anteriores (ver (3.11)). En las celdas imaginarias el valor de  $B$  vendrá dado tal y como se explica en la subsección 3.2.1.

$$B(i, j) = \frac{\sin \theta_{(j-1)_e} A_\phi(i, (j-1)_e) - \sin \theta_{j_e} A_\phi(i, j_e)}{\cos \theta_1 - \cos \theta_0} + \frac{1}{\sin(\theta)} \frac{A_\theta((i-1)_e, j) - A_\theta(i_e, j)}{\phi_1 - \phi_0} \quad (3.12)$$

Para  $i \in [1, nph + 1]$ ,  $j \in [1, nth + 1]$ .

### 3.3. Integración mediante el método de Euler explícito

Para la integración numérica de las ecuaciones del vector potencial (3.3) y (3.4), en el código original se aplica el método de Euler explícito (orden uno) [11], el cual se va a introducir brevemente.

Dado el problema de valor inicial:

$$\begin{cases} y'(t) = f(t, y(t)), & y \in \mathbb{R}^m, t \in [t_0, T], \\ y(t_0) = y_0, \end{cases} \quad (3.13)$$

el método de Euler explícito calcula el valor de  $y_1$  que aproxima a  $y(t_0 + h)$ , solución exacta de (3.13) con condiciones iniciales  $(t_0, y_0)$  y paso de integración  $h$ . Este algoritmo de resolución del PVI viene dada por:

$$\begin{cases} y_{n+1} = y_n + hf(t_n, y_n), & n = 0, 1, \dots, (T - t_0)/h, \\ y(t_0) = y_0. \end{cases}$$

En este modelo,  $y = \begin{pmatrix} A_\theta \\ A_\phi \end{pmatrix}$  y la función derivada  $f$  tiene por componentes (3.3)-(3.4). Además la integración se realiza día a día (en intervalos de 86400 segundos) durante 11 años para poder evaluar los resultados obtenidos en cada una de las posibles fases de un ciclo solar. Así se garantiza la evaluación en todos aquellos puntos  $t = t_i$  que representan los días de emergencia en los que hay que considerar las deltas de Dirac.

Una vez más, en el caso de  $A_\theta$  se trabaja con matrices de dimensión  $(nph + 1) \times nth$ , es decir,  $181 \times 90$ , y en el caso de  $A_\phi$  con matrices de dimensión  $nph \times (nth + 1)$ , es decir,  $180 \times 91$ . Por tanto, la dimensión del sistema de ecuaciones diferenciales ordinarias es 32670.

El esquema resolutivo global sigue el siguiente proceso en cada paso:

1. Se toman los valores de  $A_\theta$  y  $A_\phi$  y  $B$  en el paso anterior o de la condición inicial en caso de ser el primer paso (ecuaciones (3.6)-(3.8)).
2. Se toma el valor de  $B$  en las celdas imaginarias (ver 3.2.1).
3. Se utiliza el método de diferencias finitas para obtener los valores de  $\frac{\partial A_\theta}{\partial t}$  y  $\frac{\partial A_\phi}{\partial t}$  (ver (3.9) y (3.10)).
4. Se actualiza el vector potencial por el método de Euler explícito con paso  $h$ :

$$\begin{pmatrix} A_\theta \\ A_\phi \end{pmatrix} = \begin{pmatrix} A_\theta \\ A_\phi \end{pmatrix} + h \begin{pmatrix} \frac{\partial A_\theta}{\partial t} \\ \frac{\partial A_\phi}{\partial t} \end{pmatrix}$$

5. Avanza el tiempo:  $t = t + h$ .
6. De nuevo mediante el método de las diferencias finitas, se actualiza el valor de  $B$  a partir de los valores que se acaban de obtener de  $A_\theta$  y  $A_\phi$  (ecuación (3.11)).

Se puede concluir que la implementación de este método es sencilla, siendo la mayor complicación del código la evaluación de la función, y que en principio su coste computacional (medido en el número de evaluaciones de la función por paso) es mínimo. Sin embargo, se trata de un método con limitaciones ya que para que el error sea pequeño, los pasos han de ser pequeños también, y esto eleva el coste computacional. Además, existe la posibilidad de que los errores de redondeo acaben por contaminar los resultados obtenidos. Por este motivo, es conveniente aumentar el orden del método aplicado para la resolución de este problema.

### 3.4. Implementación de métodos Runge-Kutta de orden dos a paso fijo

En esta sección se detalla la implementación de métodos Runge-Kutta de orden dos y dos etapas para integrar numéricamente el sistema de ecuaciones diferenciales y se compararán los resultados obtenidos con el método de Euler explícito.

Dado un problema de valor inicial (3.13), su resolución mediante un método Runge-Kutta explícito de dos etapas y orden dos con paso de integración  $h$  [11], viene dada por:

$$\begin{cases} g_1 = f(t_n, y_n) \\ g_2 = f(t_n + c_2 h, y_n + h a_{21} g_1) \\ y_{n+1} = y_n + h (b_1 g_1 + b_2 g_2) \end{cases} \quad n = 0, 1, \dots, (T - t_0)/h$$

Cuya tabla de Butcher es

$$\begin{array}{c|cc} 0 & & \\ c_2 & a_{21} & \\ \hline & b_1 & b_2 \end{array}$$

Así, para resolver el modelo sobre el que se está trabajando, se seguirá el siguiente esquema en cada paso:

1. Se toman los valores de  $A_\theta$ ,  $A_\phi$  y  $B$  en el paso anterior o de la condición inicial en caso de ser el primer paso (ecuaciones (3.6)-(3.8)).
2. Se guardan estos valores de  $A_\theta$  y  $A_\phi$  como  $\hat{A}_\theta$  y  $\hat{A}_\phi$ .
3. Se toma el valor de  $B$  en las celdas imaginarias (ver 3.2.1).
4. Se utiliza el método de diferencias finitas para obtener los valores de  $\frac{\partial A_\theta}{\partial t}$  y  $\frac{\partial A_\phi}{\partial t}$  (ver (3.9) y (3.10)).
5. Se guardan estos valores de  $\frac{\partial A_\theta}{\partial t}$  y  $\frac{\partial A_\phi}{\partial t}$  como  $\frac{\partial \hat{A}_\theta}{\partial t}$  y  $\frac{\partial \hat{A}_\phi}{\partial t}$ .
6. Se actualiza el vector potencial:

$$\begin{pmatrix} A_\theta \\ A_\phi \end{pmatrix} = \begin{pmatrix} \hat{A}_\theta \\ \hat{A}_\phi \end{pmatrix} + h a_{21} \begin{pmatrix} \frac{\partial \hat{A}_\theta}{\partial t} \\ \frac{\partial \hat{A}_\phi}{\partial t} \end{pmatrix}$$

7. De nuevo mediante el método de las diferencias finitas, se actualiza el valor de  $B$  a partir de los valores que se acaban de obtener de  $A_\theta$  y  $A_\phi$  (ecuación (3.11)).
8. Se vuelven a obtener los valores de  $\frac{\partial A_\theta}{\partial t}$  y  $\frac{\partial A_\phi}{\partial t}$  utilizando la actualización de  $B$  dada por el paso anterior (ver 3.9 y 3.10).
9. Se actualiza el vector potencial siguiendo el esquema del nuevo método:

$$\begin{pmatrix} A_\theta \\ A_\phi \end{pmatrix} = \begin{pmatrix} \hat{A}_\theta \\ \hat{A}_\phi \end{pmatrix} + h b_1 \begin{pmatrix} \frac{\partial \hat{A}_\theta}{\partial t} \\ \frac{\partial \hat{A}_\phi}{\partial t} \end{pmatrix} + h b_2 \begin{pmatrix} \frac{\partial A_\theta}{\partial t} \\ \frac{\partial A_\phi}{\partial t} \end{pmatrix}$$

10. Finalmente, se actualiza el valor de  $B$  según los valores del vector potencial que se acaban de calcular (ecuación (3.11)).

**MÉTODO DEL TRAPECIO.**

El método del trapecio viene definido por:

$$\begin{cases} g_1 = f(t_n, y_n) \\ g_2 = f(t_n + h, y_n + hg_1) \\ y_{n+1} = y_n + \frac{h}{2}(g_1 + g_2), \quad n = 0, 1, \dots, (T - t_0)/h, \end{cases}$$

cuya tabla de Butcher es

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array}$$

Con este método, puede obtenerse mediante la técnica de pares encajados una estimación del error local del método de Euler [11], dada por

$$e(t_n) = \frac{h}{2} \left( \left\| \frac{\partial A_\theta}{\partial t} - \frac{\partial \hat{A}_\theta}{\partial t} \right\|_1 + \left\| \frac{\partial A_\phi}{\partial t} - \frac{\partial \hat{A}_\phi}{\partial t} \right\|_1 \right) \quad (3.14)$$

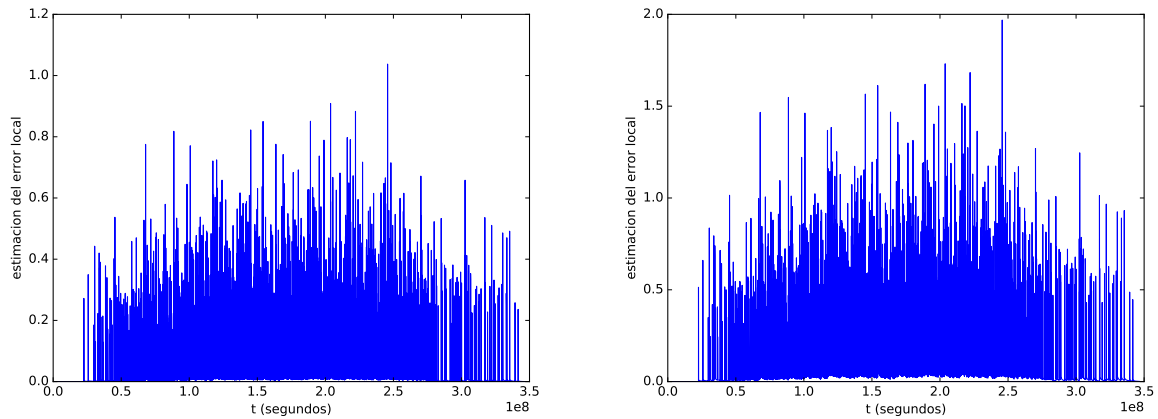


Figura 3.3: Método del trapecio. Error local estimado, 17 pasos por día (izda.), 9 pasos por día (dcha.). La magnitud de la solución es del orden de  $10^3$ .

Se puede observar en la figura 3.3 que inicialmente la estimación del error local es muy pequeña, y en el primer día de emergencia hay un salto y la estimación del error local aumenta notablemente. Estos aumentos se van sucediendo a lo largo de los 11 años ( $3,47 \times 10^8$  segundos), y cada uno de ellos se produce en los días  $t_i$  en los que hay emergencia.

Según los datos proporcionados, el primer día de emergencia es el 262 ( $0,226 \times 10^8$  segundos), que es cuando aparece el primer aumento de la estimación del error: pasa de  $8,20 \times 10^{-6}$  a  $2,70 \times 10^{-1}$ . El segundo día de emergencia es el 299 y en este caso el valor de la estimación del error aumenta de  $7,01 \times 10^{-4}$  a  $3,49 \times 10^{-1}$ . Puede observarse que el máximo se obtiene en el día 2845 con un valor de 1,03.

Por otra parte, es evidente que el tiempo que tarda el código en ejecutarse es aproximadamente el doble del que tardaba mediante el método de Euler explícito, pero desde un punto de vista computacional este método será más favorable, ya que tiene orden dos. Si se reduce el número de pasos por día a



9, el tiempo de ejecución del código vuelve a ser similar al inicial, en el que se aplicaba el método de orden uno. Sin embargo, la estimación del error local aumenta, tal y como se muestra en la figura 3.3 (dcha.).

### MÉTODO DEL PUNTO MEDIO.

El método del punto medio viene definido por:

$$\begin{cases} g_1 = f(t_n, y_n) \\ g_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}g_1) \\ y_{n+1} = y_n + hg_2, \end{cases} \quad n = 0, 1, \dots, (T - t_0)/h,$$

definido por la tabla de Butcher

$$\begin{array}{c|c} 0 & \\ \hline 1/2 & 1/2 \\ \hline 0 & 1 \end{array}$$

Con este método, puede obtenerse una estimación del error local del método de Euler, dada por

$$e(t_n) = h \left( \left\| \frac{\partial A_\theta}{\partial t} - \frac{\partial \hat{A}_\theta}{\partial t} \right\|_1 + \left\| \frac{\partial A_\phi}{\partial t} - \frac{\partial \hat{A}_\phi}{\partial t} \right\|_1 \right) \quad (3.15)$$

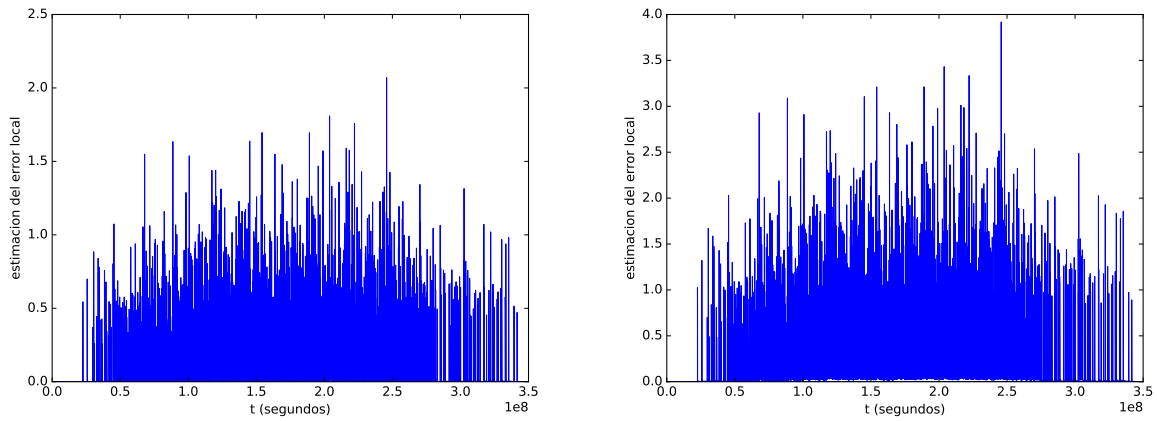


Figura 3.4: Método del punto medio. Error local estimado, 17 pasos por día (izda.). 9 pasos por día (dcha.).

Las figuras 3.4 representan el valor de la estimación local del método de Euler con 17 y 9 pasos por día respectivamente. Se observa que los resultados obtenidos con este método son menos satisfactorios que los obtenidos por el método del trapecio, pues la estimación del error local llega a alcanzar 2,07 con 17 pasos por día o 3,92 con 9 pasos por día.

El análisis llevado a cabo en este capítulo muestra las limitaciones de los métodos de paso fijo, y por ello será conveniente implementar métodos de paso variable.

### 3.5. Implementación de métodos Runge-Kutta de orden dos a paso variable

En esta subsección se implementarán los métodos del trapecio y del punto medio, pero esta vez a paso variable. Esto significa que ahora se permite que la longitud de un paso pueda variar al recorrer el intervalo de integración de longitud 1 día (86400 segundos) de modo que la estimación del error local sea siempre menor o igual que una tolerancia ( $TOL$ ) fijada por el usuario y así adaptarse al comportamiento de la solución.

$$e(t_n) \leq TOL = TOL_a + (\|A_\theta\|_1 + \|A_\phi\|_1)TOL_r \quad (3.16)$$

siendo  $TOL_a$  y  $TOL_r$  las tolerancias absoluta y relativa empleadas.

Tanto si el paso es aceptado ( $e(t_n) \leq TOL$ ) como rechazado ( $e(t_n) > TOL$ ) se emplea la siguiente fórmula para la longitud del siguiente paso de integración

$$h_{n+1} = FAC \, h_n \left( \frac{TOL}{e(t_n)} \right)^{1/2}. \quad (3.17)$$

Se toman  $FAC = 0,9$ ,  $TOL_a = 10^{-1}$  y  $TOL_r = 10^{-2}$ ,  $TOL_r = 10^{-4} \times TOL_a$ , obteniéndose los siguientes resultados:

Cuadro 3.1: Resultados numéricos obtenidos a paso variable.

$TOL_a$	TRAPECIO			PUNTO MEDIO		
	P. aceptados	P. fallados	Máx. est. error local	P. aceptados	P. fallados	Máx. est. error local
$10^{-1}$	29717	2159	$7,47 \times 10^{-1}$	30092	2730	$7,00 \times 10^{-1}$
$10^{-2}$	60761	3225	$7,40 \times 10^{-2}$	61278	3705	$7,30 \times 10^{-2}$

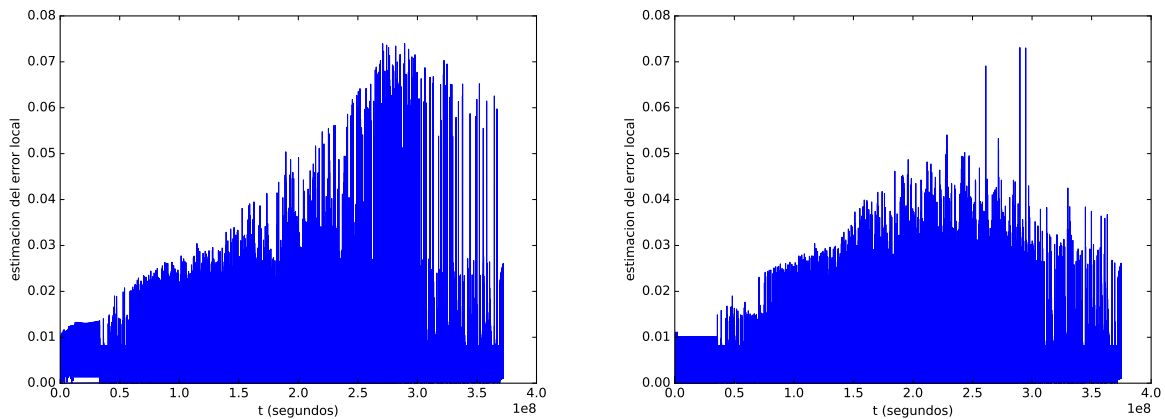


Figura 3.5: Método del trapecio (izda.), método del punto medio (dcha.). Error local estimado, paso variable.

El número de pasos que realiza el método de integración a paso fijo (en torno a 65.000) es aproximadamente el mismo que el que realiza el método de integración a paso variable con tolerancia  $10^{-1}$ , lo que significa que, mediante la implementación de un método a paso variable, se ha conseguido reducir la estimación del error local desde 1,30 hasta 0,75 en el caso del trapecio y desde 2,07 hasta 0,70 para la regla del punto medio.

### 3.6. Implementación de métodos Runge-Kutta de orden tres a paso variable

Dado el problema de valor inicial (3.13), su resolución mediante un método Runge-Kutta explícito de orden 3 viene dado por:

$$\begin{cases} g_1 = f(t_n, y_n) \\ g_2 = f(t_n + c_2 h, y_n + h a_{21} g_1) \\ g_3 = f(t_n + c_3 h, y_n + h(a_{31} g_1 + a_{32} g_2)) \\ y_{n+1} = y_n + h(b_1 g_1 + b_2 g_2 + b_3 g_3), \end{cases} \quad n = 0, 1, \dots, (T - t_0)/h,$$

cuya tabla de Butcher viene dada por

$$\begin{array}{c|ccc} 0 & & & \\ c_2 & a_{21} & & \\ c_3 & a_{31} & a_{32} & \\ \hline & b_1 & b_2 & b_3 \end{array}$$

Se considerará el método Runge-Kutta explícito de 3 etapas y orden tres dado por los coeficientes  $c_2 = 1/2$ ,  $c_3 = 1$ ,  $a_{21} = 1/2$ ,  $a_{31} = -1$ ,  $a_{32} = 2$ ,  $b_1 = 1/6$ ,  $b_2 = 4/6$  y  $b_3 = 1/6$ .

En la tabla 3.2 se muestran los resultados obtenidos en función de la tolerancia, incluyendo la estimación del error local para el método encajado del trapecio de orden 2 (dos primeras etapas). En primer lugar, se observa que cuando la tolerancia se divide entre 10, la estimación del error local también lo hace. Esto se conoce con el nombre de proporcionalidad de la tolerancia.

Cuadro 3.2: Resultados obtenidos con el método de orden 3 a paso variable.

TOL	PASOS FALLADOS	PASOS ACEPTADOS	MÁXIMO EST. ERROR LOCAL
$10^{-1}$	1881	19441	$6,56 \times 10^{-1}$
$10^{-2}$	4471	22727	$7,56 \times 10^{-2}$
$10^{-3}$	5233	29971	$7,53 \times 10^{-3}$
$10^{-4}$	6484	51537	$7,56 \times 10^{-4}$

En este caso, se puede observar de las tablas 3.1 y 3.2 que la eficiencia del método de orden tres es muy superior a cualquiera de los métodos de orden dos considerados. Para una tolerancia dada, el número de pasos requerido por el método de orden tres es claramente inferior al método de orden dos.

Las figuras 3.6 muestran la estimación del error local en todos los pasos (izda.) y sólo en los aceptados (dcha.) cuando la tolerancia vale  $TOL_a = 10^{-2}$ . Para contrastar el valor de la estimación del error local de los pasos aceptados con la de los pasos fallados, el color rojo representa la estimación del error local en los pasos fallados, y el azul en los aceptados.

La figura 3.8 muestra una comparación entre la eficiencia de los métodos de orden dos a paso variable (trapecio y punto medio) y el método de orden tres y tres etapas. El eje de ordenadas representa el error global de  $B$  en el punto final (habiendo tomado como solución exacta la obtenida con una tolerancia muy pequeña) y el eje de abscisas representa el costo computacional medido en número de evaluaciones de la función derivada. Salvo para tolerancias mayores a  $10^{-1}$ , el método de orden 3 es claramente superior a los métodos de orden dos, ya que su costo computacional es menor para obtener un mismo error. De los métodos implementados, el Runge-Kutta de orden tres es el más adecuado para el problema considerado en esta memoria.

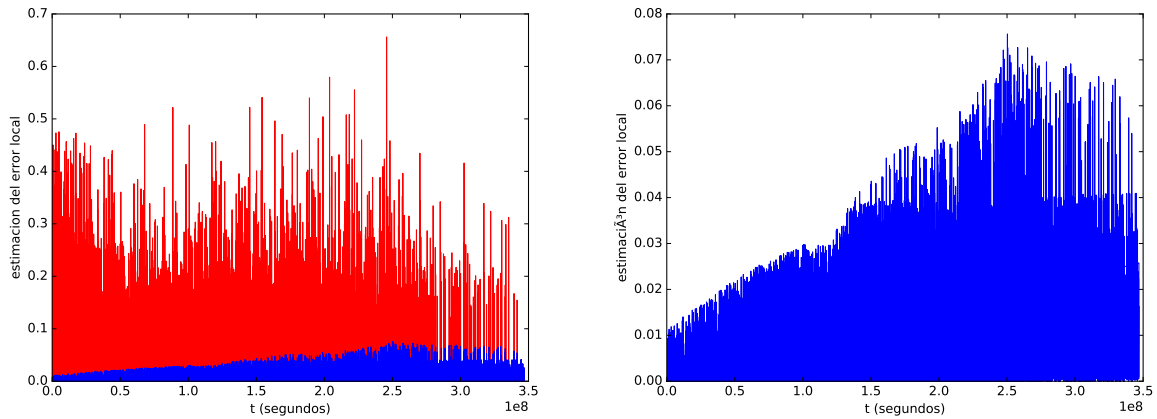


Figura 3.6: Estimación error local, pasos aceptados (azul), pasos fallados (rojo).  $TOL_a = 10^{-2}$ .

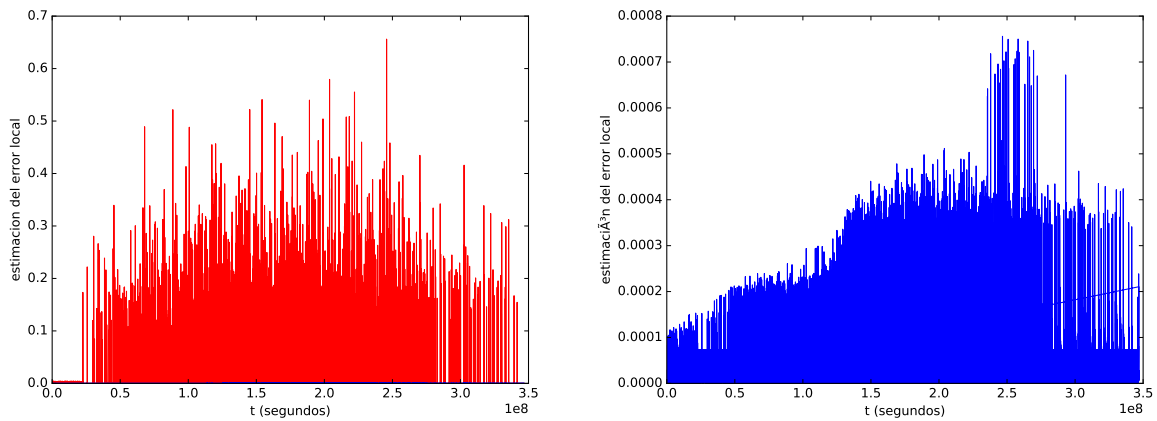


Figura 3.7: Estimación del error local, pasos aceptados (azul), pasos fallados (rojo).  $TOL_a = 10^{-4}$ .

### 3.7. Resultados proporcionados por el código

El código con el que se ha trabajado guarda una serie de imágenes de la intensidad del campo magnético  $B$  en la superficie solar (figura 3.9) según la estructura del mallado 3.2. El color rojo y el azul hacen referencia a las polaridades del campo magnético. El hecho de que estos colores se encuentren muy concentrados en la parte superior e inferior del gráfico muestra la estructura dipolar del campo magnético solar en cada instante de tiempo. Las manchas que aparecen en el centro de la imagen tienen relación con la aparición de regiones magnéticas dipolares emergentes. Este conjunto de imágenes son útiles para el estudio de la inversión de polaridad que se produce en el Sol.

En la figura 3.10 se muestra una segunda imagen impresa por el código. En la gráfica superior está el diagrama de mariposa se observa la inversión de polaridad en el Sol. En la parte izquierda de este diagrama, el color rojo se encuentra en la parte alta y el azul en la parte baja, pero a la derecha estas posiciones están invertidas. La gráfica central muestra la intensidad normalizada total del campo magnético en función del tiempo y en la última, cada curva representa la intensidad del campo magnético en uno de los polos geográficos del Sol en función del tiempo. El signo de las curvas indica la polaridad. Notar que en torno al cuarto año hay un cambio de polaridad que se observa en la primera gráfica y en el cruce de las curvas de la última gráfica. También es el punto en el que la intensidad del campo magnético es máxima (gráfica central).

Si se omiten las emergencias de nuevas regiones dipolares (figura 3.11), se observa que la intensidad del campo magnético decae (gráfica central) y no hay inversión de la polaridad.

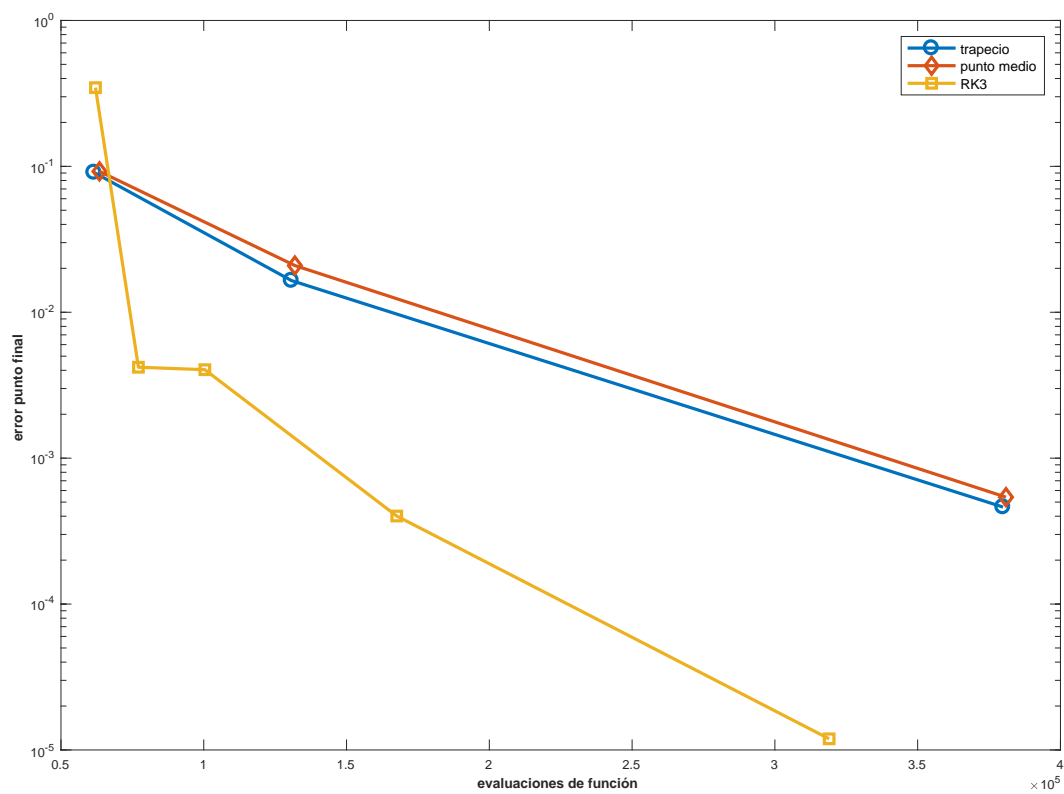


Figura 3.8: Gráfica de eficiencia de los métodos Runge-Kutta implementados a paso variable.

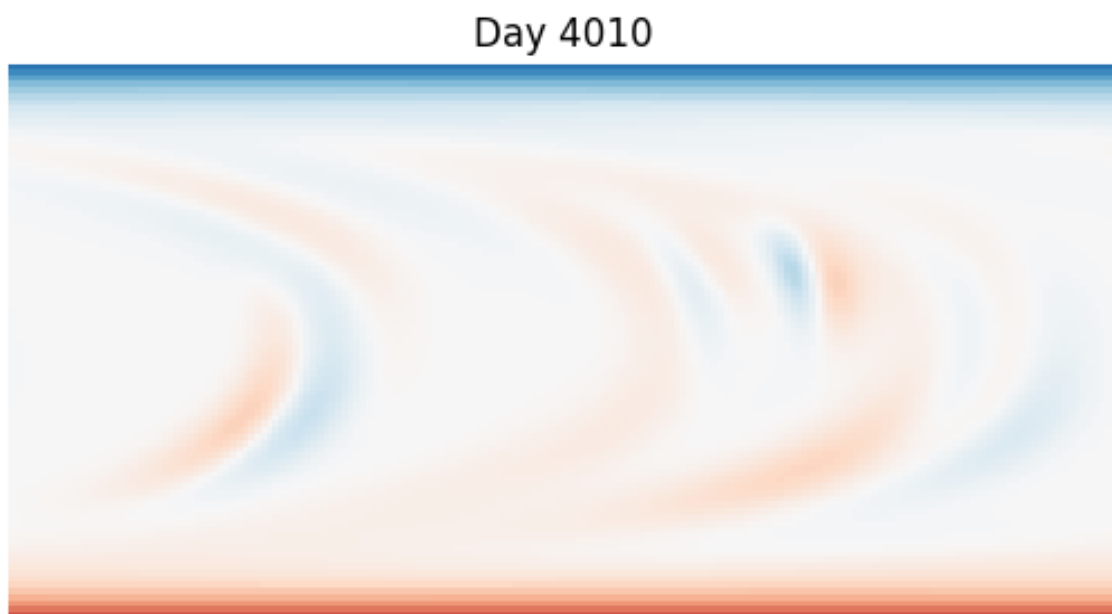


Figura 3.9: Imagen de la intensidad del campo magnético y su polaridad en el día 4010 en la superficie solar.

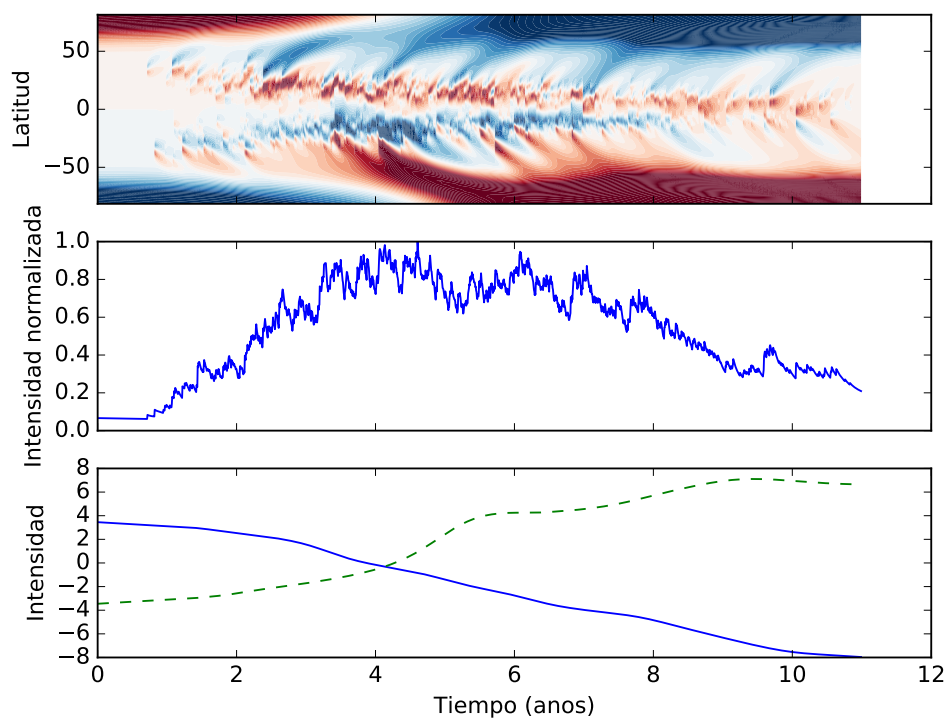


Figura 3.10: Diagrama de Mariposa y representación de la intensidad del campo magnético con emergencia.

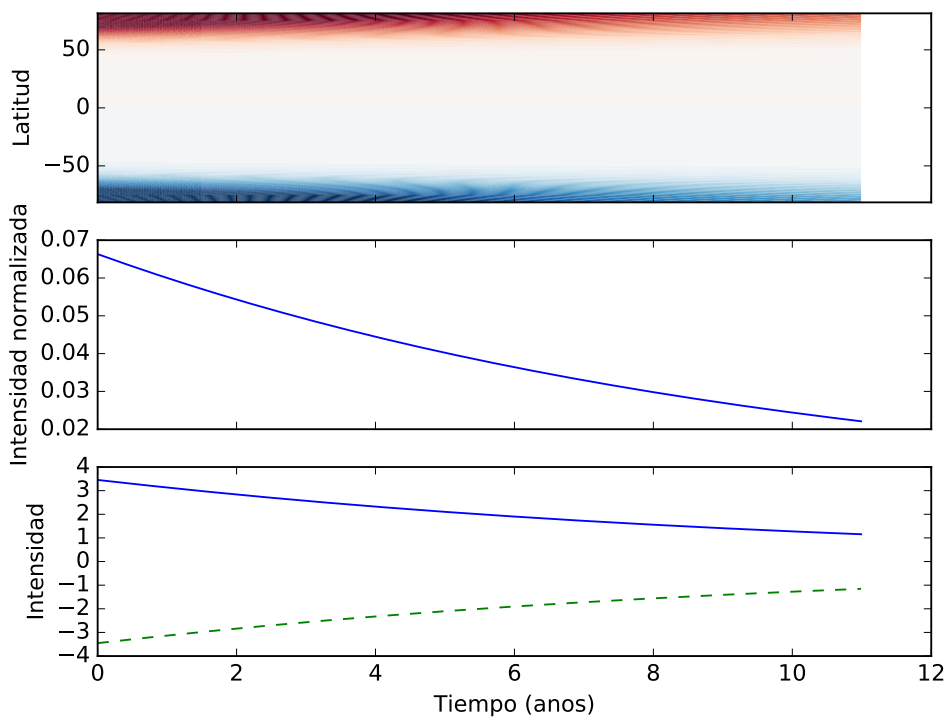


Figura 3.11: Diagrama de Mariposa y representación de la intensidad del campo magnético sin emergencia.

## Capítulo 4

# Conclusiones

En este Trabajo de Fin de Grado se ha analizado un código en Python proporcionado por el Dr. Yeates para la resolución numérica de la ecuación de transporte de flujo magnético en la superficie solar. Se trata de una ecuación en derivadas parciales que, tras ser semidiscretizada espacialmente (método de diferencias finitas) da lugar a un sistema de ecuaciones diferenciales ordinarias con deltas de Dirac. En este código se realizaba la integración numérica mediante el método de Euler explícito, pero no se tenía ningún conocimiento cualitativo de los resultados obtenidos.

Se han implementado sobre el código dos métodos de orden dos, que permitieron calcular una estimación del error local que cometía el método de Euler explícito. Se observó que esta estimación del error local daba lugar a valores altos y que los métodos a paso fijo podían no ser suficientemente adecuados. Entonces, se implementaron estos mismos métodos a paso variable, de forma que el paso de integración se adaptara para hacer que los valores de la estimación del error local fueran inferiores a una cierta tolerancia. Se consiguió de esta forma, con el mismo costo computacional, una estimación del error local más pequeña a la obtenida a paso fijo (desde 1,30 hasta 0,75 en el caso del trapecio y desde 2,07 hasta 0,70 para la regla del punto medio).

Análogamente, se implementó un método Runge-Kutta de 3 etapas y orden 3 a paso variable. Este esquema es claramente superior a los de orden 2 anteriores, en particular con el mismo costo computacional que el código original, el error global es como mínimo 10 veces menor.

### 4.1. Dificultades encontradas

- Trabajar en el campo de la astrofísica y del magnetismo solar me ha resultado complicado, ya que no tenía ningún conocimiento previo en la materia. Sin embargo, ha sido interesante poder introducirme entre estos conceptos para así poder entender la interpretación de los resultados y el modelo.
- El hecho de que la ecuación contuviera un sumatorio con deltas de Dirac ha hecho que el problema a resolver (en particular la evaluación de la función y su implementación en el código) fuese complejo.
- Una de las mayores dificultades a la hora de desarrollar el trabajo fue partir de un código escrito por otra persona, adaptarme a la nomenclatura, estructura y evaluación de la función derivada de la ecuación diferencial, para posteriormente implementar sobre él los diferentes métodos utilizados, así como realizar el cambio de paso fijo a paso variable de forma óptima.

## 4.2. Continuidad y mejoras del trabajo

Los códigos y métodos expuestos en este trabajo realizan la integración día a día. Esto significa que, sea cual sea la longitud de los pasos de integración, se para en cada día volviéndose a reiniciar la integración. Se planteó la idea de cambiar este hecho y hacer que el método parase únicamente en los días  $t_i$  de emergencia dados por la base de datos del Dr. Yeates. Sin embargo, la implementación de un código que cumpliera esto es más compleja, ya que para obtener los diagramas finales se necesita información de todos los días, habría que aplicar métodos continuos para aproximar los valores en cada día. Tras analizar los resultados obtenidos por un código de estas características, se observó que no merecía la pena debido a la complejidad del código, y a que los resultados fueron similares.

Resolver la ecuación con un método de orden superior podría no ser conveniente debido al alto costo computacional y a que el orden de la semidiscretización espacial utilizada es bajo.



# Bibliografía

- [1] R.B. LEIGHTON. TRANSPORT OF MAGNETIC FIELDS ON THE SUN, *The Astrophysical Journal*, vol. 140, 140:1547, November 1964
- [2] H.W. BABCOCK. THE TOPOLOGY OF THE SUN'S MAGNETIC FIELD AND THE 22-YEAR CYCLE, *The Astrophysical Journal*, vol. 133, 572:587, Marzo 1961
- [3] C.J. SCHRIJVER, I. BAUMANN. MODELING THE SUN'S OPEN MAGNETIC FLUX, *Astronomy and Astrophysics Journal*, Diciembre 2006
- [4] PAUL CHARBONNEAU, CROSS-HEMISPHERIC COUPLING IN A BABCOCK-LEIGHTON MODEL OF THE SOLAR CYCLE, *Advances in Space Research*, vol. 40, 899:906, Enero 2007
- [5] J.JIANG, D.H.HATHAWAY, R.H.CAMERON, S.K.SOLANKI, L.GIZON, L.UPTON, MAGNETIC FLUX TRANSPORT AT THE SOLAR SURFACE, *Space Science Reviews Journal*, Septiembre 2014
- [6] A.R.YEATES, CORONAL MAGNETIC FIELD EVOLUTION FROM 1996 TO 2012: CONTINUOUS NON-POTENCIAL SIMULATIONS, *Solar Phys Journal*, Abril 2013
- [7] A.R.YEATES, D.BAKER, L. VAN DRIEL-GESZTELYI, SOURCE OF A PROMINENT POLEWARD SURGE DURING SOLAR CYCLE 24, *Solar Phys Journal*, Febrero 2015
- [8] MARKO NEDELJKOV, MICHAEL OBERGUGGENBERGER, ORDINARY DIFFERENTIAL EQUATIONS WITH DELTA FUNCTION TERMS, *Publications de l'Institut Mathématique, Nouvelle série*, tome 91(105), 125:135, 2012
- [9] HAIRER, ERNST, NØRSETT, SYVERT P., WANNER, GERHARD, SOLVING ORDINARY DIFFERENTIAL EQUATIONS I, NONSTIFF PROBLEMS. SECOND REVISED EDITION. SPRINGER, Berlin, 1993
- [10] [https://www.nasa.gov/mission\\_pages/sunearth/news/solarcycle-primer.html](https://www.nasa.gov/mission_pages/sunearth/news/solarcycle-primer.html)
- [11] <https://solarsystem.nasa.gov/planets/sun/indepth>
- [12] [https://www.nasa.gov/mission\\_pages/sunearth/spaceweather/index.html](https://www.nasa.gov/mission_pages/sunearth/spaceweather/index.html)



# ANEXO

## **CÓDIGO DE INTEGRACIÓN** **MÉTODO RUNGE KUTTA DE ORDEN 3 Y 3 ETAPAS A PASO VARIABLE.**

```
# Simple surface flux transport model
# - python version: Agosto 2017
#
#
#
from __future__ import division
import numpy as np
import matplotlib.pyplot as plt
from bmr import BMR
from numpy import linalg as LA

# (0) Simulation parameters
# -----
nth = 90          # number of grid points in latitude
nph = 180         # number of grid points in longitude
ndays = 365*11    # number of days to run simulation
bflyFreq = 2      # frequency to output butterfly diagram [in days]
diagFreq = 2      # frequency to output fluxes [in days]
bipFile = 'bips.txt' # name of the bipole data file
B0 = 10.0         # initial polar field strength
latPole = 70.0*np.pi/180.0 # location of pole boundary for calculating polar flux
movie = True
#movie = False

movfreq = 27
bmax = 30.0 # Plotting threshold (G) for running map of Br

# (1) Read in the BMR list from file [probably neater way]
# -----
rsun = 6.96e10
bmrs = [];
f=open(bipFile,'r')
data = f.readlines()
f.close()
nbips = int(data[0].split()[0])
print('Reading in '+str(nbips)+' BMRs from '+bipFile)
for k in range(1,nbips+1):
```

```

data1 = data[k].split()
bday1 = int(float(data1[0]))
blon1 = float(data1[1])*np.pi/180.0
blat1 = float(data1[2])*np.pi/180.0
bsep1 = float(data1[3])
bflux1 = float(data1[4])/rsun**2
btilt1 = float(data1[5])*np.pi/180.0
bmrs.append(BMR(bday1,blon1,blat1,bsep1,bflux1,btilt1))
f.close()
nxtbmr = 0    # Initialise pointer to next BMR

# (2) Set up the grid (equally spaced in phi and cos(theta))
# -----
# (a) Arrays at cell centres
cthc1 = (np.arange(nth)+0.5)/nth*2.0 - 1.0
phc1 = (np.arange(nph)+0.5)/nph*2*np.pi
# Make into 2-dimensional arrays:
phc, cthc = np.meshgrid(phc1,cthc1,indexing='ij')
thc = np.arccos(cthc)
latc1 = np.arcsin(cthc1)*180.0/np.pi
sthc = np.sin(thc)
# (b) Grid spacings:
dph = phc1[1] - phc1[0]
dcth = cthc1[1] - cthc1[0]
# (c) Arrays on theta-ribs:
temp = np.arange(nph+1)
temp, ctht = np.meshgrid(temp,cthc1,indexing='ij')
tht = np.arccos(ctht)
stht = np.sin(tht)
# (d) Arrays on phi-ribs:
cthp1 = (np.arange(nth + 1))/float(nth)*2.0 - 1.0
temp, cthp = np.meshgrid(phc1,cthp1,indexing='ij')
thp = np.arccos(cthp)
sthp = np.sin(thp)

# (3) Initialise flows and diffusion
# -----
# (a) Supergranular diffusion coefficient:
eta = 500.0/6.96e5**2          # i.e. 500 km^2/s
# (b) Differential rotation (on th ribs):
omA = (13.7195 - 360.0/27.2753)/180.0*np.pi/86400.0
omB = -2.396/180.0*np.pi/86400.0
omC = -1.787/180.0*np.pi/86400.0
om = omA + omB*ctht**2 + omC*ctht**4
vph_om = om*stht
# (c) Meridional flow (on ph ribs):
# -- Schuessler-Baumann profile
v0 = 16.0/6.96e8              # i.e. 11 m/s
latp = 0.5*np.pi - thp
vth_mf = -np.sin(2.0*latp)*np.exp(np.pi-2*np.abs(latp))
v0 = v0/np.max(np.abs(vth_mf))

```

```

vth_mf = vth_mf*v0
vth_mf[:,0] = 0.0
vth_mf[:,nth] = 0.0

# (d) Exponential flux decay time:
tdec = 10.0*86400.0*365.25
invtdec = 1.0/tdec

# (4) Set timestep for CFL condition
# -----
cflFactor = 0.2
# Minimum grid spacings:
hphmin = np.min(np.abs(sthc*dph))
hthmin = np.min(np.abs(thp[0,1:nth]-thp[0,0:nth-1]))
# Maximum timestep for diffusion term:
dt_eta = np.min([hphmin**2/eta,hthmin**2/eta])
# Maximum timestep for meridional flow:
t_mf = np.abs((thp[:,1:nth]-thp[:,0:nth-1])/vth_mf[:,0:nth-1])
dt_mf = np.min(t_mf)
# Maximum timestep for differential rotation:
dt_om = np.min(np.abs(stht*dph/vph_om))
dt = min([dt_eta, dt_mf, dt_om])*cflFactor
print('Chosen timestep (s):'+str(dt))
# Modify dt to fit evenly into a day:
ndt = int(round(86400.0/dt))
ndt=9
dt = 86400.0/ndt
print('Rounded timestep (s):'+str(dt))

# (5) Initial condition
# -----
# Choose initial distribution of Br.
br = np.zeros((nph+2,nth+2)) # array includes ghost cells (182*92)
latc = 0.5*np.pi - thc
brNew = latc/np.abs(latc)*np.exp(-np.exp(np.pi - 2*np.abs(latc))*v0/4.0/eta*(np.sin(2*np.abs(latc))))
br[1:nph+1,1:nth+1] = brNew/np.max(np.abs(brNew))*B0

# Compute corresponding vector potential. Since
# Br = 1/(sin(th))*( d/dth(sin(th)Aph) - d/dph(Ath) )
# we can choose
# Aph(th,ph) = 0 [ on ph ribs ]
# Ath(th,ph) = -sin(th)*int_0^ph Br(ph',th) dph' [ on th ribs ]
aph = np.zeros((nph,nth+1))
ath = np.zeros((nph+1,nth))
ath[1:nph+1,:] = -sthc*np.cumsum(br[1:nph+1,1:nth+1],axis=0)*dph

# (6) Initialise output arrays
# -----
t = 0.0
# Initialise butterfly diagram:

```

```

nbfly = ndays//bflyFreq
bfly = np.zeros((nbfly+1,nth))
tbfly = np.zeros((nbfly+1))
bfly[0,:] = np.mean(br[1:nph+1,1:nth+1],axis=0)
tbfly[0] = t
ibfly = 1
# Initialise total fluxes:
ndiag = ndays//diagFreq
totflux = np.zeros(ndiag+1)
nflux = np.zeros(ndiag+1)
sflux = np.zeros(ndiag+1)
tdiag = np.zeros(ndiag+1)
br[1:nph+1,1:nth+1] = (sthp[:,0:nth]*aph[:,0:nth] - sthp[:,1:nth+1]*aph[:,1:nth+1])/dcth +
totflux[0] = np.sum(np.abs(br[1:nph+1,1:nth+1]))*dph*dcth
maskNPole = ( thc <= (0.5*np.pi - latPole) )
maskSPole = ( thc >= (0.5*np.pi + latPole) )
nflux[0] = np.sum(br[1:nph+1,1:nth+1]*maskNPole)*dph*dcth
sflux[0] = np.sum(br[1:nph+1,1:nth+1]*maskSPole)*dph*dcth
tdiag[0] = t
idiag = 1
xab= np.zeros(200000)
yab= np.zeros(200000)
index=0
pasos_fallados=0
pasos_aceptados=0
# (7) Main loop
# -----
if movie:
plt.ion()
plt.set_cmapp('RdBu_r')
mov = plt.imshow(np.transpose(br[1:nph+1,1:nth+1]),vmin=-bmax,vmax=bmax)

for d in range(1,ndays+1):
if ((d%100)==0):
print('Day '+str(d)+' of '+str(ndays))
# (a) Add in new regions if necessary
while(bmrs[nxtbmr].day==d):
ath[1:nph+1,:] = ath[1:nph+1,:] + bmrs[nxtbmr].ath(phc, thc)
nxtbmr += 1

# print t, k, d
#
#=====
# Se calcula la primera etapa antes del bucle para
# aprovechar en el caso de paso fallado
#
# valores iniciales Y0
#
ath0 = ath; aph0 = aph
# (b) Apply boundary conditions
# (i) periodic in ph

```

```

br[0,1:nth+1] = br[nph,1:nth+1]
br[nph+1,1:nth+1] = br[1,1:nth+1]
# (ii) closed in th (i.e., no diffusion through boundary)
br[1:nph+1,0] = br[1:nph+1,1]
br[1:nph+1,nth+1] = br[1:nph+1,nth]

# (c) Compute RHS on ribs from differential rotation
dat = vph_om*0.5*(br[0:nph+1,1:nth+1] + br[1:nph+2,1:nth+1])

# (d) Compute RHS on ribs from meridional flow
dap = -vth_mf*0.5*(br[1:nph+1,0:nth+1] + br[1:nph+1,1:nth+2])

# (e) Compute RHS on ribs from supergranular diffusion
dat = dat - eta/stht*(br[1:nph+2,1:nth+1] - br[0:nph+1,1:nth+1])/dph
dap = dap - eta/sthp*(br[1:nph+1,1:nth+2] - br[1:nph+1,0:nth+1])/dcth

# (f) Compute RHS on ribs from exponential flux decay
dat = dat - ath*invtdc
dap = dap - aph*invtdc
#
# G1 es (dat1, dap1)
#
dat1 = dat; dap1 = dap;
#=====
t0 = 0;
tf = 86400 # 24*3600 = 86400 segundos
dt = 86400.0/(ndt) # paso inicial
tol = 1e-4;
tolr = 1e-4
fallo=0

while t0 < tf:
# (g) Update vector potential
ath = ath + dt/2*dat
aph = aph + dt/2*dap
#
# Y1 = Y0 + h*G1
#
# (h) Advance time
t = t + dt/2
# (i) Update br from vector potential:
# Br = 1/(sin(th))*( d/dth(sin(th)Aph) - d/dph(Ath) )
# = -d/d(cos(th))( sin(th)*Aph ) - (1/sin(th))*d/dph( Ath )
br[1:nph+1,1:nth+1] = (sthp[:,0:nth]*aph[:,0:nth] - sthp[:,1:nth+1]*aph[:,1:nth+1])/dcth +
#
#
# Calculo de G2, segunda evaluacion de la funcion
#
#
br[0,1:nth+1] = br[nph,1:nth+1]

```

```

br[nph+1,1:nth+1] = br[1,1:nth+1]
# (ii) closed in th (i.e., no diffusion through boundary)
br[1:nph+1,0] = br[1:nph+1,1]
br[1:nph+1,nth+1] = br[1:nph+1,nth]

# (c) Compute RHS on ribs from differential rotation
dat = vph_om*0.5*(br[0:nph+1,1:nth+1] + br[1:nph+2,1:nth+1])

# (d) Compute RHS on ribs from meridional flow
dap = -vth_mf*0.5*(br[1:nph+1,0:nth+1] + br[1:nph+1,1:nth+2])

# (e) Compute RHS on ribs from supergranular diffusion
dat = dat - eta/stht*(br[1:nph+2,1:nth+1] - br[0:nph+1,1:nth+1])/dph
dap = dap - eta*sthp*(br[1:nph+1,1:nth+2] - br[1:nph+1,0:nth+1])/dcth

# (f) Compute RHS on ribs from exponential flux decay
dat = dat - ath*invtddec
dap = dap - aph*invtddec

dat2=dat
dap2=dap
#
# G2=(dat2, dap2)

# (g) Update vector potential
ath = ath0 - dt*dat1 +2*dt*dat2
aph = aph0 - dt*dap1 +2*dt*dap2

# Update br from vector potential:
# Br = 1/(sin(th))*( d/dth(sin(th)Aph) - d/dph(Ath) )
# = -d/d(cos(th))( sin(th)*Aph ) - (1/sin(th))*d/dph( Ath )
br[1:nph+1,1:nth+1] = (sthp[:,0:nth]*aph[:,0:nth] - sthp[:,1:nth+1]*aph[:,1:nth+1])/dcth +
#
#
# Calculo de G3, tercera evaluacion de la funcion
#
#
t = t + dt/2
br[0,1:nth+1] = br[nph,1:nth+1]
br[nph+1,1:nth+1] = br[1,1:nth+1]
# (ii) closed in th (i.e., no diffusion through boundary)
br[1:nph+1,0] = br[1:nph+1,1]
br[1:nph+1,nth+1] = br[1:nph+1,nth]

# (c) Compute RHS on ribs from differential rotation
dat = vph_om*0.5*(br[0:nph+1,1:nth+1] + br[1:nph+2,1:nth+1])

# (d) Compute RHS on ribs from meridional flow
dap = -vth_mf*0.5*(br[1:nph+1,0:nth+1] + br[1:nph+1,1:nth+2])

# (e) Compute RHS on ribs from supergranular diffusion

```



```

dat = dat - eta/stht*(br[1:nph+2,1:nth+1] - br[0:nph+1,1:nth+1])/dph
dap = dap - eta*sthp*(br[1:nph+1,1:nth+2] - br[1:nph+1,0:nth+1])/dcth

```

```

# (f) Compute RHS on ribs from exponential flux decay

```

```

dat = dat - ath*invtdc

```

```

dap = dap - aph*invtdc

```

```

#

```

```

# G3=(dat, dap)

```

```

#

```

```

# (g) Update vector potential  $Y_{n+1}=Y_n-(1/6)hg_1+(4/6)hg_2+(1/6)hg_3$ 

```

```

ath = ath0 + dt/6*(dat1+4*dat2+dat)

```

```

aph = aph0 + dt/6*(dap1+4*dap2+dap)

```

```

# estimacion del error local del metodo de Euler explicito

```

```

#

```

```

"""

```

```

xab[index] = t

```

```

est = dt*( LA.norm( dat-dat1,1) + LA.norm( dap-dap1, 1))

```

```

est1=est

```

```

if est > tol*(1+ (LA.norm( ath0,1) + LA.norm( aph0, 1))/100):

```

```

est1 = -est

```

```

yab[index] = est1

```

```

index = index+1

```

```

print t, est1, d, dt/9600, pasos_aceptados, pasos_fallados

```

```

"""

```

```

est = (dt/6)*( LA.norm( dat1-(2*dat2)+dat,1) + LA.norm(dap1-(2*dap2)+dap,1))

```

```

#if yab[index-1] > 0.1:

```

```

#   ioioio = input('ojo')

```

```

xab[index] = t

```

```

yab[index] = -est

```

```

index = index+1

```

```

if est <= tol*(1+ (LA.norm( ath0,1) + LA.norm( aph0, 1))*tolr):

```

```

yab[index-1]=-yab[index-1]

```

```

#if est>0.075:

```

```

# print t, est, d, dt/9600, pasos_aceptados, pasos_fallados, (LA.norm( ath0,1) + LA.norm( ap

```

```

#if d==4015:

```

```

#print pasos_aceptados, pasos_fallados

```

```

# (i) Update br from vector potential:

```

```

# Br = 1/(sin(th))*( d/dth(sin(th)Aph) - d/dph(Ath) )

```

```

# = -d/d(cos(th))( sin(th)*Aph ) - (1/sin(th))*d/dph( Ath )

```

```

br[1:nph+1,1:nth+1] = (sthp[:,0:nth]*aph[:,0:nth] - sthp[:,1:nth+1]*aph[:,1:nth+1])/dcth +

```

```

ath0 = ath; aph0 = aph

```

```

# (b) Apply boundary conditions

```

```

# (i) periodic in ph

```

```

br[0,1:nth+1] = br[nph,1:nth+1]

```

```

br[nph+1,1:nth+1] = br[1,1:nth+1]
# (ii) closed in th (i.e., no diffusion through boundary)
br[1:nph+1,0] = br[1:nph+1,1]
br[1:nph+1,nth+1] = br[1:nph+1,nth]

# (c) Compute RHS on ribs from differential rotation
dat = vph_om*0.5*(br[0:nph+1,1:nth+1] + br[1:nph+2,1:nth+1])

# (d) Compute RHS on ribs from meridional flow
dap = -vth_mf*0.5*(br[1:nph+1,0:nth+1] + br[1:nph+1,1:nth+2])

# (e) Compute RHS on ribs from supergranular diffusion
dat = dat - eta/stht*(br[1:nph+2,1:nth+1] - br[0:nph+1,1:nth+1])/dph
dap = dap - eta*sthp*(br[1:nph+1,1:nth+2] - br[1:nph+1,0:nth+1])/dcth

# (f) Compute RHS on ribs from exponential flux decay
dat = dat - ath*invtdc
dap = dap - aph*invtdc
#
# G1 es (dat1, dap1)
#
dat1=dat; dap1=dap;
t0 = t0 + dt
factor=0.9*min( 4.0, ((tol/(est+1e-15))*(1/3)) )
if fallo:
factor = min(1.0, factor)
dt = factor*dt
fallo=0
pasos_aceptados=pasos_aceptados+1
if t0+dt > tf:
dt = tf - t0
else:
t = t - dt      # modifcacion en los otros esquemas
pasos_fallados=pasos_fallados+1
dt = 0.9*max( 0.1, ((tol/est)*(1/3)) )*dt
ath = ath0; aph = aph0
fallo=1

#=====
#
# FIN BUCLE EN LOS ndt DIAS
#
#=====

# (j) Running movie of Br
if (movie & (d%movfreq==0)):
mov.set_data(np.transpose(br[1:nph+1,1:nth+1]))
plt.title('Day %g' % d)
plt.draw()

# Save butterfly diagram if it is time:

```

```

if ((d%bflyFreq)==0):
    bfly[ibfly,:] = np.mean(br[1:nph+1,1:nth+1],axis=0)
    tbfly[ibfly] = t/86400.0
    ibfly += 1

# Save total fluxes if it is time:
if ((d%diagFreq)==0):
    totflux[idiag] = np.sum(np.abs(br[1:nph+1,1:nth+1]))*dph*dcth
    nflux[idiag] = np.sum(br[1:nph+1,1:nth+1]*maskNPole)*dph*dcth
    sflux[idiag] = np.sum(br[1:nph+1,1:nth+1]*maskSPole)*dph*dcth
    tdiag[idiag] = t/86400.0
    idiag += 1

# (8) Plot output
# -----
if movie:
    plt.ioff()

f, ax = plt.subplots(3, sharex=True)
plt.set_cmap('RdBu_r')
#
# Butterfly diagram:
ax[0].contourf(tbfly/365.25,latc1,bfly.transpose(),128,vmin=-6,vmax=6)
ax[0].set_ylabel('Latitud')
#
# Total flux:
ax[1].plot(tdiag/365.25, totflux/179.8985313678109)
ax[1].set_ylabel('Intensidad normalizada')
#
# Polar fluxes:
ax[2].plot(tdiag/365.25, nflux)
ax[2].plot(tdiag/365.25, sflux,'--')
ax[2].set_ylabel('Intensidad')
ax[2].set_xlabel('Año')
plt.show()

plt.figure()
plt.plot(xab,np.maximum(yab,0))
plt.xlabel('t (segundos)')
plt.ylabel(u'estimación del error local')
plt.show()

plt.figure()
plt.plot( xab, -np.minimum(yab,0),'r',xab, np.maximum(yab,0),'b')
plt.xlabel('t (segundos)')
plt.ylabel(u'estimacion del error local')
plt.show()

```

